AD-A067 166   L N K CORP SILVER SPRING MD   F/G 9/4
THE USE OF MODELS IN IMAGE ANALYSIS.(U)
JAN 79 G C STOCKMAN, S H KOPSTEIN   F33615-76-C-0521

UNCLASSIFIED   AMRL-TR-78-117   NL

1 OF 2

AD
A067166

2

B.S.

AMRL-TR-78-117

LEVEL II

THE USE OF MODELS IN IMAGE ANALYSIS

GEORGE C. STOCKMAN
STEVEN H. KOPSTEIN
L.N.K. CORPORATION
302 NOTLEY COURT
SILVER SPRING, MARYLAND 20904

JANUARY 1979

Approved for public release; distribution unlimited.

AEROSPACE MEDICAL RESEARCH LABORATORY
AEROSPACE MEDICAL DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICES

Please do not request copies of this report from Aerospace Medical Research Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22161

Federal Government agencies and their contractors registered with Defense Documentation Center should direct requests for copies of this report to:

Defense Documentation Center
Cameron Station
Alexandria, Virginia 22314

TECHNICAL REVIEW AND APPROVAL

AMRL-TR-78-117

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

HENNING E. VON GIERKE
Director
Biodynamics and Bioengineering Division
Aerospace Medical Research Laboratory

AIR FORCE/56780/16 March 1979 — 150

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19 REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
18 AMRL-TR-78-117

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER
9

4. TITLE (and Subtitle)
6 THE USE OF MODELS IN IMAGE ANALYSIS

5. TYPE OF REPORT & PERIOD COVERED
Technical Report,
1 Nov 76-31 Aug 78

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
10 George C. Stockman
Steven H. Kopstein

8. CONTRACT OR GRANT NUMBER(s)
15 F33615-76-C-0521 New

9. PERFORMING ORGANIZATION NAME AND ADDRESS
L.N.K. CORPORATION
302 Notley Court
Silver Spring, Maryland 20904

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
62202F, 7184-25-03
16 17 25

11. CONTROLLING OFFICE NAME AND ADDRESS
Aerospace Medical Research Laboratory, Aerospace Medical Division, Air Force Systems Command
Wright-Patterson Air Force Base, Ohio 45433

12. REPORT DATE
11 Jan [redacted] 79

13. NUMBER OF PAGES
149

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
12 149p.

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
N/A

16. DISTRIBUTION STATEMENT (of this Report)
Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
feature extraction, image processing, image understanding, interactive imagery screening, models, object detection, registration.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Several of the automatic components of an interactive imagery screening and target detection system are studied. Components include 1) a module for detection of primitive image features without model hypothesis, 2) a module for generation of hypotheses given that a particular model for the data has been evoked by primitive feature content, and 3) a module for testing the model generated hypotheses against the data. Only shape features extracted from boundary segments are considered in this work. Features used are straight, circular, and

→ next page

391718

79 04 06 018

parabolic edge segments and points of intersection or points of high curvature on edge segments. Feature extraction algorithms are discussed. Two different model types are considered--problem reduction representations (PRR) which correspond to grammar models and iconic models which correspond to cartographic data bases. Once primitive features from an image are aligned with a potential model the model generates hypotheses about the data which can be either verified or denied. Testing of hypotheses is done via curve fitting or template matching. A new approach to registration is presented and documented via experimental results. The registration approach enables the use of cartographic data bases as models for use in image analysis and object detection.

Summary

A study of the use of models in image analysis is reported. Models are structured a priori information which can be used to interpret data in a manner consistent with real-world knowledge. Potential models are selected by primitive feature extraction. Primitive features studied in this research were all derived from boundary curve segments, i.e. edges, of the image. Two types of models were considered for encoding real-world structural knowledge. One model studied was Problem Reduction Representation (PRR) or equivalently the Context Free Grammar (CFG) which generically specifies structure. The second type of model considered was the Geographic Data Base (GDB) which iconically encodes particular shape features to be seen in aerial imagery.

Whatever model is used, primitive features are required to align a hypothetical model with raw image data. Further analysis is then made by verification of structural hypotheses. Verification is treated here as either template matching or curve fitting under constraints.

The study attempted to draw conclusions about an entire image screening system by studying several possible parts. Many experiments were performed and a large amount of literature reviewed. As a result of the study, the following conclusions were reached.

- Useful interpretation of imagery requires that instances of sensed data be integrated with large amounts of stored real-world knowledge.

- Representation of real-world knowledge, particularly for use by a computer, is a difficult task with much current research activity. Generic models such as PRR or CFG are difficult to use in practice but particular iconic shape models appear to have practical potential.

- Current automatic primitive feature detection techniques can support complex analysis when features are registered to an iconic model.

- Evaluating and combining confidence values for verifying hypotheses about image structure is difficult in both theory and practice and requires further work.

- The most promising future direction for reconnaissance image analysis appears to be toward map-guided image analysis.

PREFACE

The work reported here was performed by L.N.K. CORPORATION under Contract F33615-76-C-0521. Dr. Laveen Kanal served as Principal Investigator, and Dr. George Stockman as Senior Scientist. Dr. Hans L. Oestreicher, Chief, Mathematics and Analysis Branch, Biodynamics and Bioengineering Division, Aerospace Medical Research Laboratory, Wright-Pattern Air Force Base, served as technical monitor.

Table of Contents

List of Figures

List of Tables

1. Introduction

Efficient automatic or semiautomatic analysis of aerial imagery is a problem of great practical interest to the Air Force. The complexity of the image interpretation tasks of target identification and target location has thus far been too great for automatic processing. However, success has been achieved in the recognition of simple or stereotyped objects [Ashbaugh, 1973] and in the automatic verification of certain mapped features in imagery [Barrow 1977]. The computer can outperform a human in some detection tasks and has the virtue of being indefatigueable in its efforts. It is natural then to attempt a man-machine synthesis whereby image analysis would be achieved with each component performing the tasks which it does best.

Investigation of interactive screening of reconnaissance imagery was begun by L.N.K. CORPORATION in October of 1974 under Contract F33615-75-C-5056. Preliminary results were reported in Stockman and Kanal [1976] and recommendations for future work were made. This report summarizes the results achieved during a follow-up investigation of certain subproblems broken down in the initial study. Figure 1. shows the possible flow of information and control in an interactive imagery screening system. Stations 1 and 2 are used for imagery for which no previous computer-stored analysis exists. Primitive detectors are applied at station 1 to detect features common to targets - straight edge activity, corners, parallel edges, or symmetrical edge activity. If any such features are detected the imagery is examined further at station 2 where object models are tested against the data automatically. If any targets are detected the human analyst (station 4) is alerted for further interpretation of the imagery and for compilation of a symbolic image for the data base to be used in repeat coverage at some later date. Whenever imagery is input to the system and symbolic coverage exists in the data base, an automatic process (station 3) attempts to match, or register, the new raw imagery to the symbolic imagery. Apparently, much matching can be done automatically [Barrow, 1977:Stockman,1978] at this stage. Significant discrepancies detected during the matching of data to archive must be brought to the attention of the human interpreter for further analysis.

For implementation of a system as described, blocks 1, 2, and 3 are problematical because they involve computer decision-making. This report examines possible implementations for blocks 1, 2, and 3 of the interactive screening system. Section 2 of this report deals with primitive detection; that is, with the automatic recognition of primitive image features without benefit of context or higher level knowledge. Curves, straight lines, corners, and points of high curvature are discussed as important primitives. The primitives are useful not only in block 1 as evidence of cultural activity but also in block 3 to automatically establish a correspondence (registration) between the image and a map in the archive. Section 3 discusses the use of grammar models for object recognition and section 4 considers object detection as registration of image edges to object model edges. The automatic recognition of full objects is required for successful implementation of block 2 while the registration technique developed in section 4 is useful for the registration required in block 3 of Figure 1.

Conclusions are rendered in Section 6 of this report. Very briefly, it can be said here that all practical image analysis problems require the input of information from sources other than the input imagery itself. Representation of this outside knowledge for use by an automatic process is one of the most interesting and difficult problems under current research. Perhaps the most promising current alternative is to use positional knowledge as encoded in present day cartographic data bases. By using such symbolic data bases and a system such as that in Figure 1. it should be possible to do analysis of repeat coverage much more rapidly and accurately than original coverage. In this manner, a large initial investment in human analysis can provide machine useable knowledge for future payoffs in automatic analysis.

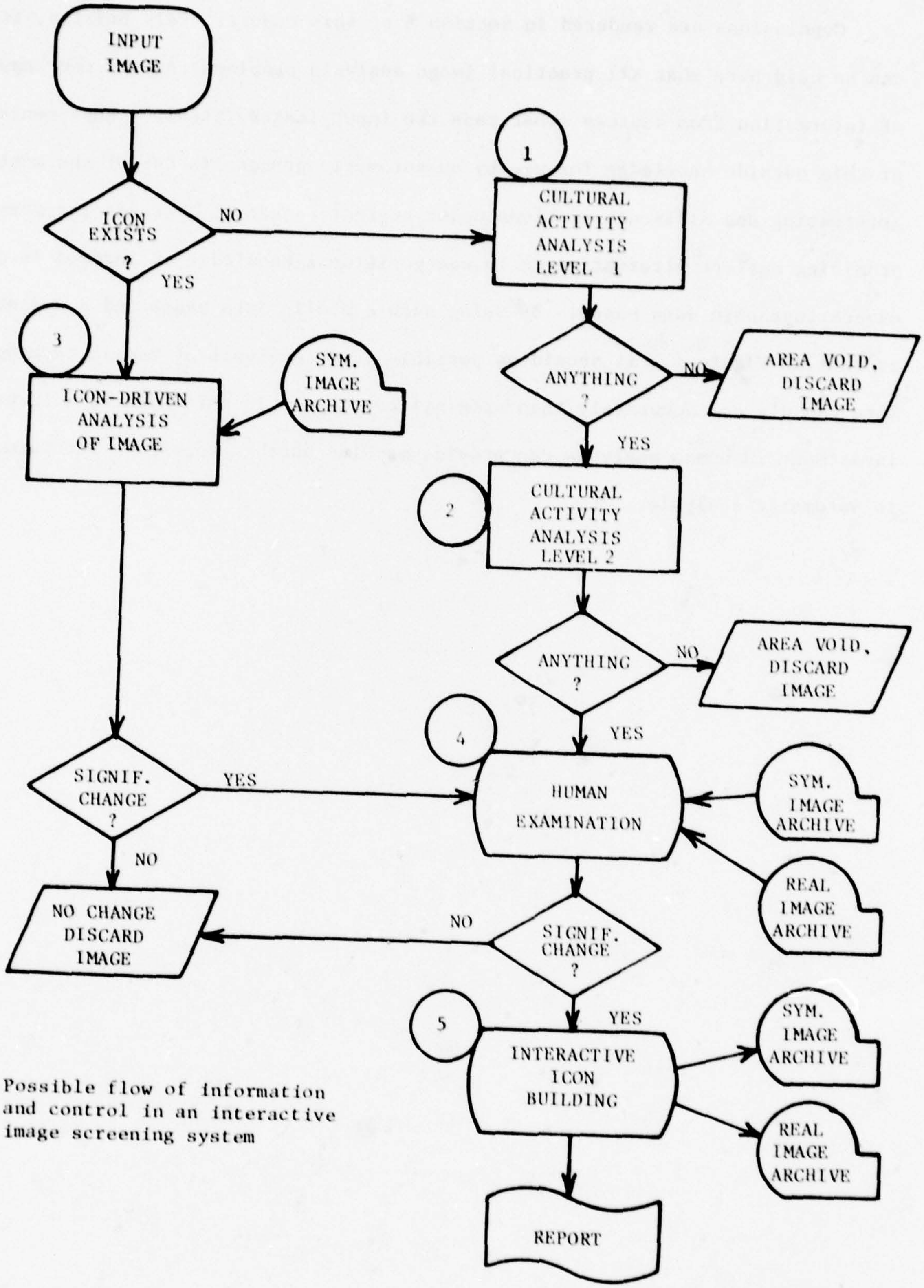


Figure 1. Possible flow of information and control in an interactive image screening system

2. Primitive detection

This section discusses the extraction of primitive features from grey scale imagery. All features used here are edge dependent features in the sense that their detection depends on detection of the boundary between two regions of contrasting grey scale. Edge primitives or edge elements are not single contrast points but rather a minimum collection of them defining a connected and continuous segment of a boundary. It is a common view [Marr 1975] that such edge elements form the basis on which higher level human recognition processes operate.

Image points of high contrast can be automatically identified by a number of mathematical techniques of varying complexity. A survey of edge detection is given in [Davis 1975] and experiments are reported in [Bullock 1974] and [Rosenfeld 1971]. After reviewing the literature on edge detection and experimenting with several techniques the author has arrived at the following conclusions.

- Due to lack of contrast, edge operators cannot be expected to extract all the edge points from any real world scene.

- Due to image noise there will be automatically extracted edge points in places where a human will not perceive them.

- The ideal edge content of any real world scene cannot generally be extracted without considerable semantic information from sources outside of the imagery.

. If a small part of the outside semantic information used by a human in image interpretation were available to an automatic process, most of the current edge detection schemes would be entirely adequate to support complete image analysis.

The fourth point made asserts an optimistic view in spite of the three initial negative remarks. It seems clear that research on primitive edge detection should be curtailed while work on semantic interpretation and use of knowledge should be pursued. It is assumed in this report that current simple edge detection operators are sufficient for capturing an essential representation of a scene. An essential representation is one that supports semantic interpretation and hypothesis formation which can in turn be used for driving more focused edge detection operations. The rest of this section discusses rather simple and economical methods for extracting partial edge content from imagery to be used for higher level interpretation. The fact that these simple operators fail in many cases does not preclude correct image analysis since higher-level operators will come into play.

2.1 Extraction of smooth edge elements

Very general boundary curve detection consists of two simple steps - - first, high contrast(edge) points are identified in the image and secondly sets of these points are organized into continuous curve segments. Due to noise and low contrast it is unreasonable to expect unbroken boundary curves as a result of such general low-level processing. Higher level processing using geometric or topological constraints can connect curve segments into complete boundary structures. Partial semantic interpretation of the image may be necessary in order to reliably connect curve segments. Features of existing curve segments such as length, curvature, degree of match to a stored prototype may be used in the interpretation and/or connection decisions. This section addresses only very general low-level curve segment extraction which is appropriate for arbitrary problem domains. Enhancement and interpretation of the curve segment set via specific semantics is the topic of future work.

Curve segments representing the image data can be extracted in a 3-step process. First of all, all image points are examined and a set of high contrast points is extracted. A Roberts' type gradient operator is applied to each point and a gradient magnitude and direction are extracted. L.N.K. has obtained good results by keeping only that 5% of image points which have the highest gradient magnitude (contrast). The second process examines a small neighborhood around each edge point extracted in step 1 and finds the best continuing edge point in the forward and backward direction. Links are set pointing to the best continuation points if such points exist in the high contrast set. These links are established independently (theoretically in parallel) for each high contrast point. The third step extracts curve segments as chains of high contrast points mutually linked together in step 2 of the procedure.

2.1.1 Step 1: extraction of high contrast edge points

L.N.K. has developed a gradient operator based on masks which allows gradient direction to be optionally computed at resolution of 1/8, 1/16, or 1/32 of the circle (i.e. 45°, 22$\frac{1}{2}$°, or 11$\frac{1}{4}$°). Gradient magnitudes are histogrammed and a fixed percentage of the highest contrast points are selected. Recent work has been done with 2%, 5%, or 10% of the image points. High contrast points are saved in array storage outside of the image storage. Appendix A documents the simple edge operator based on masks.

2.1.2 Step 2: finding continuing points by local processing

The neighborhood of each high contrast point is independently examined by a spiraling search around the given point. (see Figure 2.) Neighbors closest to the point are considered first and only neighbors within a fixed radius r are examined. Links are established to the first high contrast point with acceptable gradient direction continuing a curve in either the forward or backward direction.

The amount of curvature to be tolerated in the curve is expressed as a tolerance on the agreement of gradient directions. The forward direction of traversal of a curve is taken to be that direction of traversal placing the darker region to the right of the curve. This processing induces two relations on the set of high contrast edge points $E = \{e_1, e_2, \ldots, e_n\}$. $F = \{(e_i, e_j) : e_i$ forward links to $e_j\}$ and $B = \{(e_k, e_\ell) : e_k$ backward links to $e_\ell\}$. Note that $(e_i, e_j) \varepsilon F$ does not mean that $(e_j, e_i) \varepsilon B$. This symmetry will probably exist if edge points e_i and e_j are indeed consecutive points on the same boundary segment. However, at locations of curve junctions or poor contrast, the edge point relationships are expected to be broken. Linking of points is done in image array storage.

2.1.3 Step 3: collection of continuous chains of edge points

If $(e_i, e_j) \varepsilon F$ and $(e_j, e_i) \varepsilon B$ then e_i and e_j are consecutive points on a curve segment. All the high contrast points can now be placed into equivalence classes (representing curve segments) as follows. Define the relationship R such that $(e_i, e_j) \varepsilon R$ if and only if there is a chain of forward (backward) links (possibly a null chain) from point e_i to point e_j and a chain of backward (forward) links from point e_j to point e_i. R is relexive, symmetric and transitive. Each equivalence class represents a separate curve segment. Curve segments can then be extracted by considering each point of the high contrast set (in any order) and tracking all related points when a beginning curve point is encountered. A beginning curve point is a point e_i such that if $(e_i, e_j) \varepsilon B$ then $(e_j, e_i) \not\varepsilon F$. Tracking of curve segments is done in image array storage. The image is raster-scanned for beginning points. When a beginning point is found the chain is tracked until broken. Then the raster scan resumes at its former place. Because each curve segment has but one beginning point there is no duplication. Chains smaller than some fixed number of points are suppressed thus removing many noise edges.

2.1.4 Examples of smooth curve segment extraction

Figure 3 shows a light airplane on a darker airfield. The curve extraction procedure as applied to a window containing the right wing tip is illustrated in Figures 4 , 5 , and 6. The high contrast points near the wing tip and their gradient directions are shown in Figure 4. Figure 5 shows a plot of all forward and backward links created by the spiraling neighborhood searches of step 2 of the process. Notice that certain points are of degree 3 meaning that they are at the junctions of multiple edge activity. These points must be at the terminus of an extracted curve segment because they cannot relate symmetrically to 3 neighbors. Large sets, or chains, of symmetrically related points are shown in Figure 6. A large portion of the wing boundary is successfully extracted along with two edges of the "USAF" identification interior to the wing and two edges of a dark streak on the airfield below the plane.

Figure 7 shows a photo containing curved roads in rough terrain (lower right corner). The high contrast points from a region where two roads intersect are shown in Figure 8. The point linking relations are shown in Figure 9 and the five curve segments extracted are shown in Figure 10. The straight curve segment oriented toward 225° is caused by a shadow which cuts across one of the intersecting roads. The other curves are from the road edges and a midstrip structure at the intersection.

2.1.5 Discussion of smooth curve extraction

The curve extraction algorithm has been used to support a registration procedure which matches curve segments of an image with those of a map or model. Segments with points of high curvature were selected and measured for curvature and typed as either concave or convex. These features of the extracted curves allowed for selective matching to curves in the map or model. Many of the wing tips, tail tips, and nose tips of a set of airplanes were extracted and used for registration in this manner. Some airplane parts were missed due to a fracturing of the curves. There were similar problems with the terrain imagery due to shadows or low contrast.

Higher level problem specific knowledge must be employed to join general curve segments to form the boundary of recognizable objects. The interpretation of the curves depends on the recognition of the objects and visa versa. As a simple example, if a set of curve segments map onto parts of an airplane model under the same RS&T* transformation, an appropriate linking of curve segments for forming the continuous boundary is immediately suggested. L.N.K. has been successful at verifying faint curve segments under model direction and has thus been able to get complete boundary curves for modeled objects even when high contrast points form only a partial object boundary. More discussion on this topic follows in Section 4.

*Rotation, Scaling and Translation

```
      INTEGER DX,DY,COMPMT
      COMMON /DELTAS/DX(68),DY(68),COMPMT(68)

      DATA DX / 1,1,0,-1,-1,0,1,
     +     2,2,2,1,0,-1,-2,-2,-2,-1,0,1,
     +  2,3,3,3,2,1,0,-1,-2,-3,-3,-3,-2,-1,0,1,
     +  2,3,3,2,-2,-3,-3,-2,-1,0,1,4,4,4,1,0,
     +  -1,-4,-4,-4,-4,-3,-2,2,3,4,4,3.2,-2,-3,-4/

C                   59 45 46 47 60
C                58 44 34 35 36 37 61
C             57 43 33 18 19 20 21 38 62
C             56 32 17  6  7  8  9 22 48
C             55 31 16  5  *  1 10 23 49
C             54 30 15  4  3  2 11 24 50
C             68 42 29 14 13 12 25 39 63
C                67 41 28 27 26 40 64
C                   66 53 52 51 65
C
C
      DATA   DY /0,-1,-1,-1,0,1,1,1,
     +            1,0,-1,-2,-2,-2,-1,0,1,2,2,2,
     +  2,1,0,-1,-2,-3,-3,-3,-2,-1,0,1,2,3,3,3,
     +  3,2,-2,-3,-3,-2,2,3,4,4,4,1,0,-1,-4,-4,
     +  -4,-1,0,1,2,3,4,4,3,2,-2,-3,-4,-4,-3,-2/
      DATA COMPMT/5,6,7,8,1,2,3,4,15,16,17,18,19,20,9,10,11,12,13,14,
     +  29,30,31,32,33,34,35,36,21,22,23,24,25,26,27,28,
     +  41,42,43,44,37,38,39,40,51,52,53,54,55,56,45,46,
     +  47,48,49,50,63,64,65,66,67,68,57,58,59,60,61,62/
C COMPMT DEFINES 180 DEGREE ROTATIONS COMPMT(0) IS UNDEFINED
```

Figure 2. Definition of spiral search sequence through the neighbors of a pixel*.

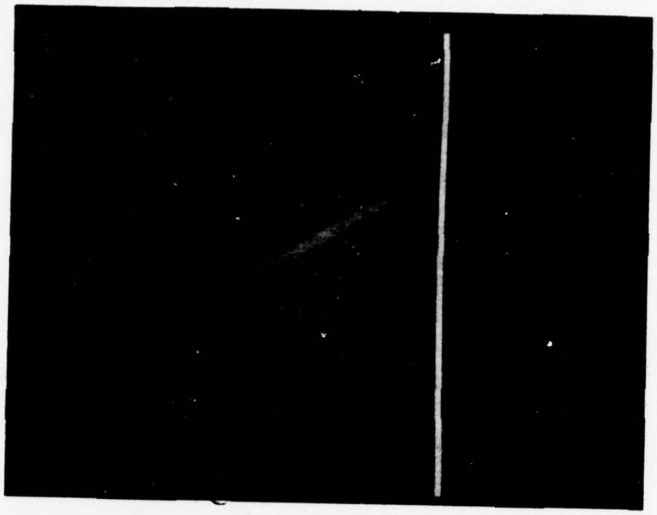


Figure 3. Airplane on airfield background.
(AFB1 test image)

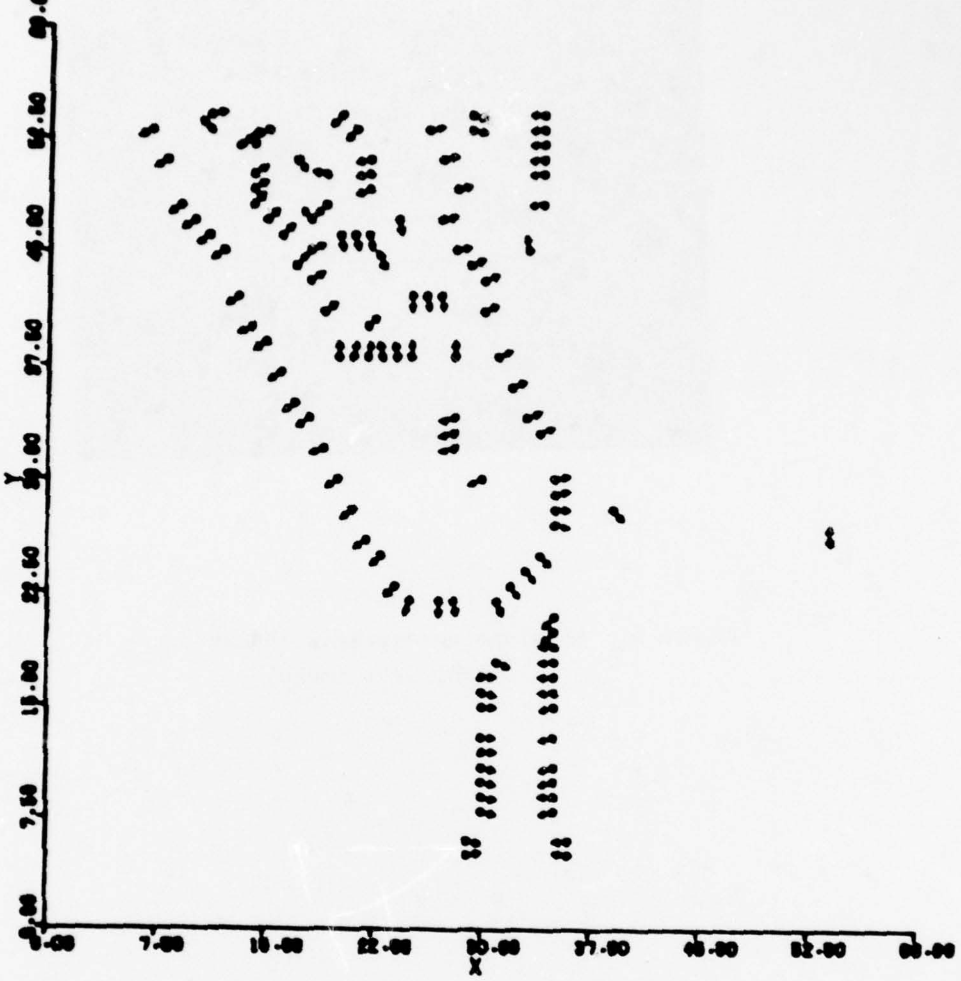


Figure 4. Gradient direction of high contrast points of right airplane wing. (Curve detection step 1.)

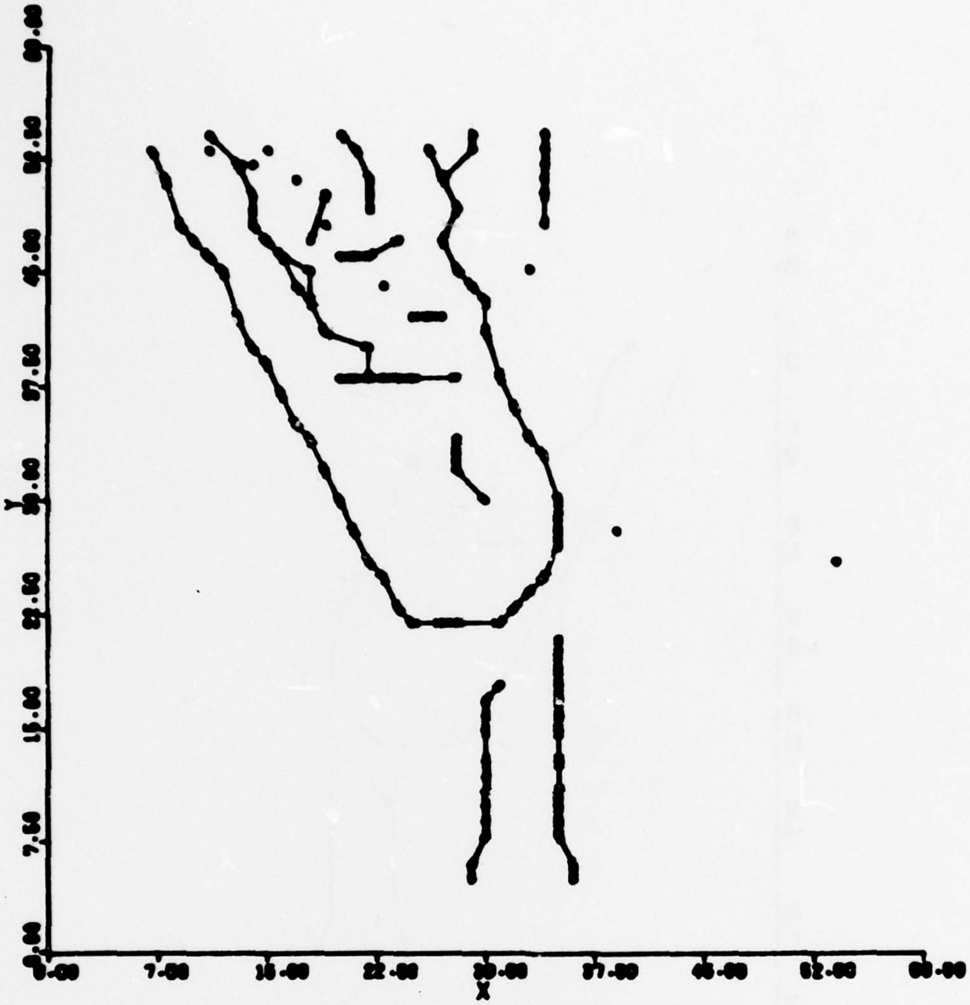


Figure 5. Plot of all forward and backward linking relationships among high contrast points of Figure 4 (curve detection step 2.)

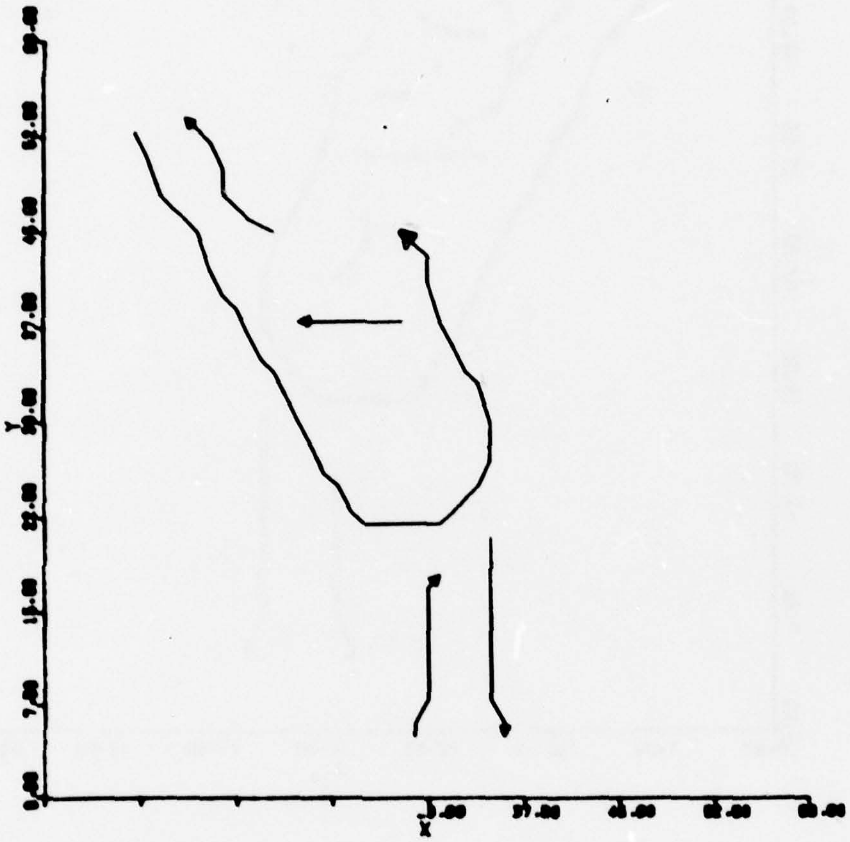


Figure 6. Long curve segments extracted from related points of Figure 5 (curve detection step 3.)

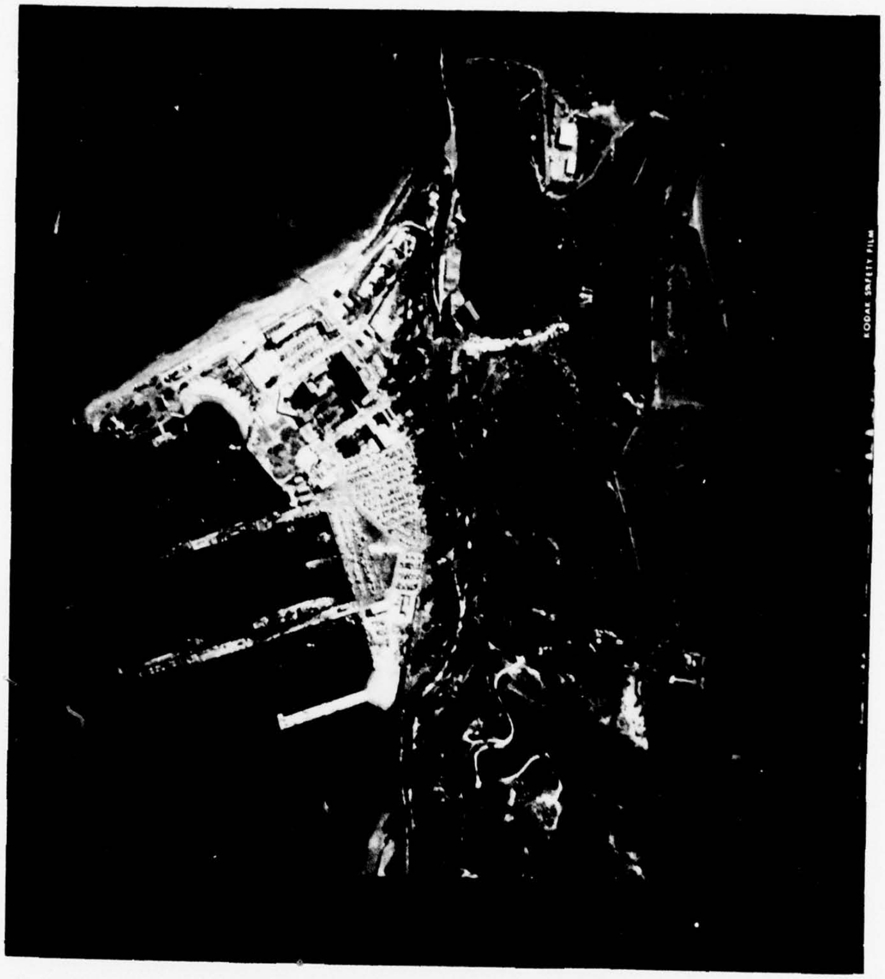


Figure 7. Naval base photo containing network of curving roads.(UNB2 test image)

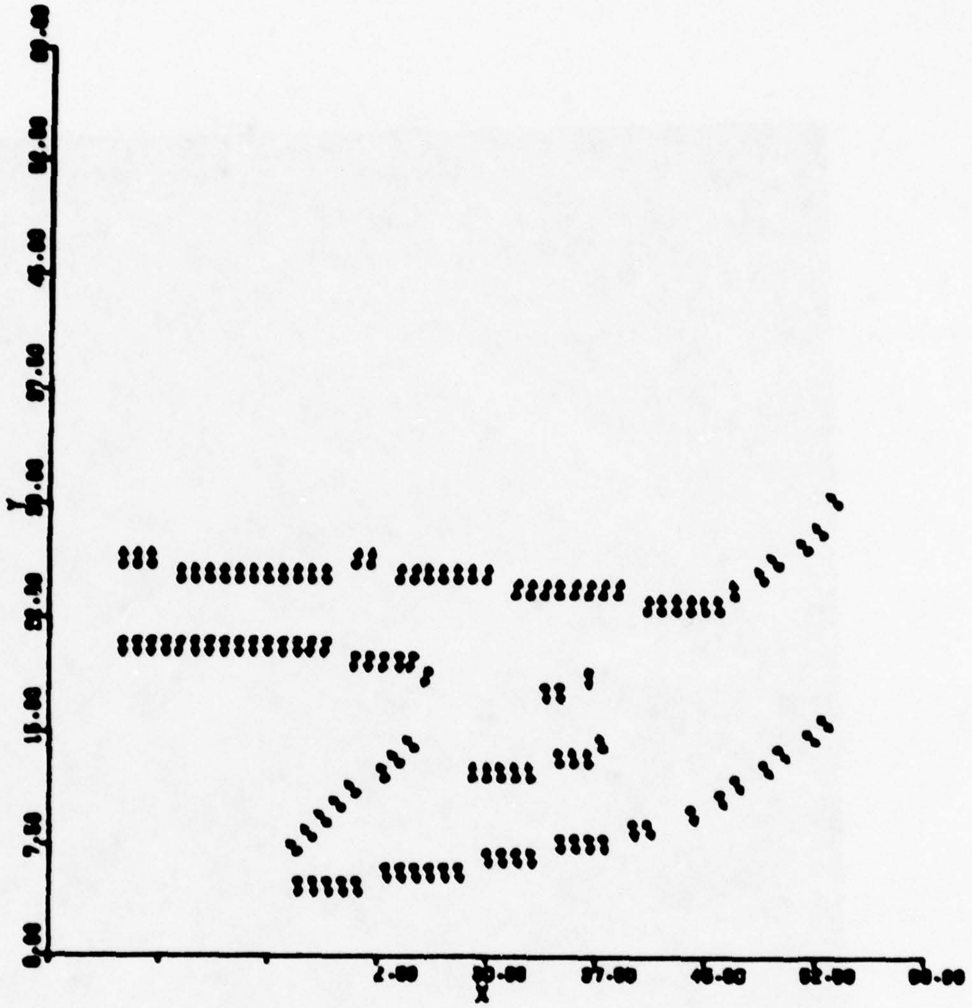


Figure 8. Gradient direction of high contrast points on road intersection in Figure 7.

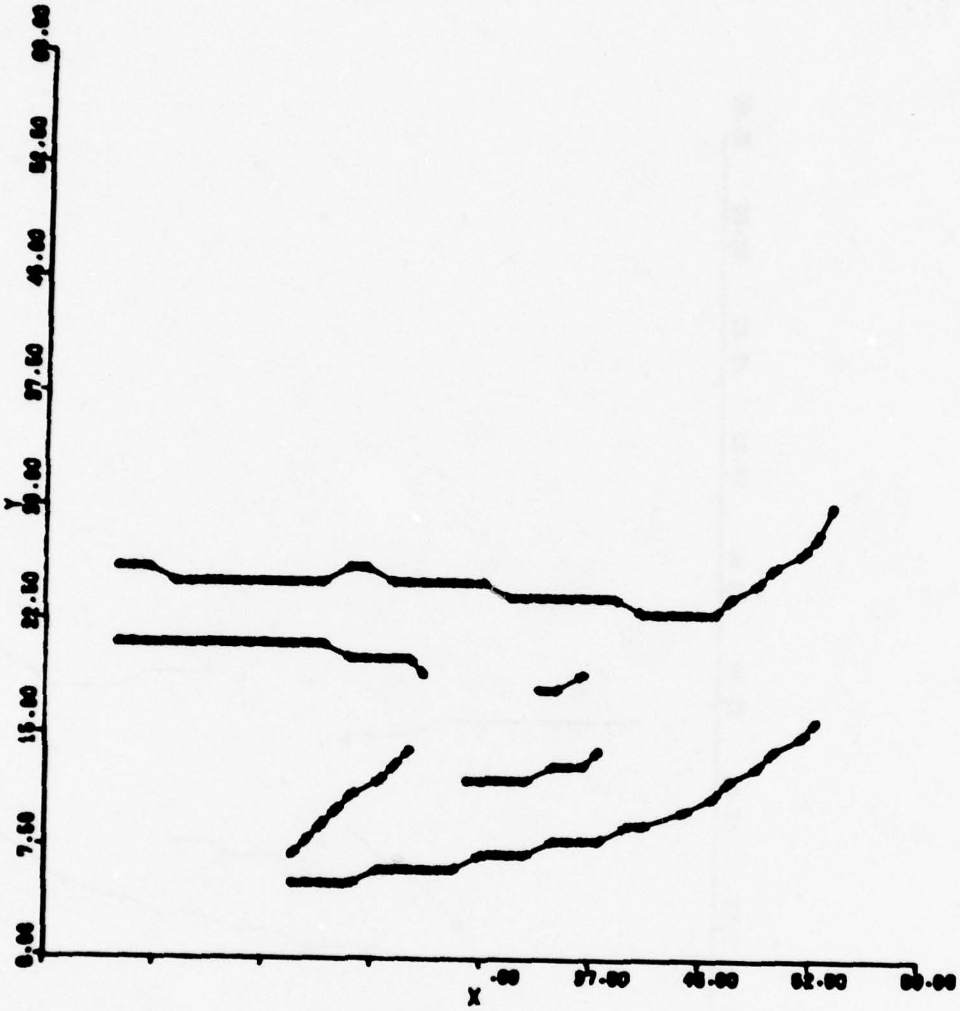


Figure 9. Plot of all forward and backward linking relationships among high contrast points of Figure 8.

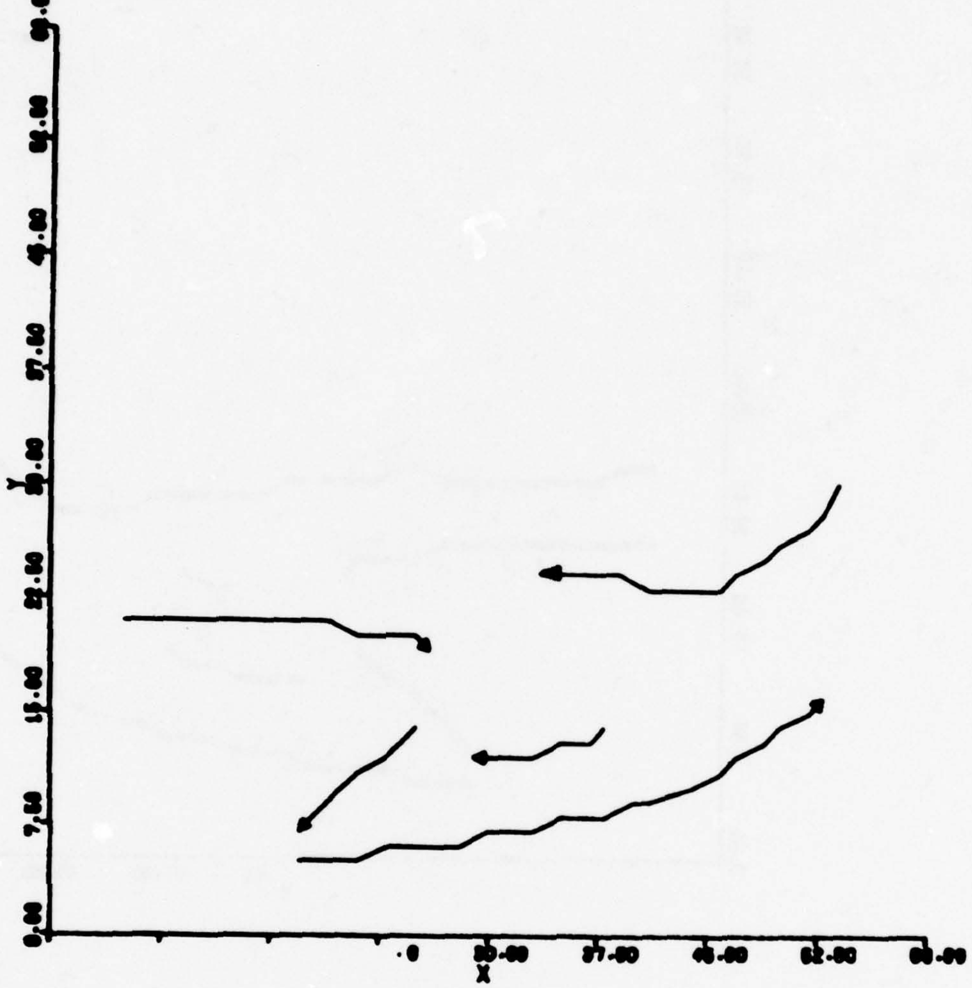


Figure 10. Long curve segments extracted from related points of Figure 9.

2.2 Extraction of straight edge elements

The Hough transformation is an efficient device for detecting if a set of high contrast points are organized along a mathematical curve [Duda 1972]. The simplest mathematical curve and the most important one for detection of man-made structures is the straight line. Only two parameters are required for specification of a given line - - in polar form the parameters are the direction of the normal to the line (θ) and the distance from the origin to the line (r). If the possible line directions are discretized to T values and the possible distances from the origin are discretized to R values then Hough detection is logically equivalent to a matching of T·R templates to the high gradient points [Stockman 1977].

2.2.1 Enhancements to the general Hough transform

Three special enhancements were made by L.N.K. in its use of the Hough transform. First of all, only a small percentage of the high gradient points were passed to the Hough detector. This was achieved by histogramming the gradient image and setting a selection threshold such that 2%, 5%, or 10% of the image points were passed. An exception to this policy occured if the threshold were lower than an estimate of the standard deviation of noise gradients in uniform regions. In that case the threshold was set to the noise level gotten from interactive training on uniform regions of imagery. By using only a small percentage of the strongest edge points, only the strongest edges would be detected while weak edges or noise edges would be suppressed. This had the effect of making the false alarm rate for detections almost 0 while the false dismissal rate was high.

A second enhancement made by L.N.K. to the general Hough line detector was the provision for refining the resolution of the parameters of detected lines. Coarse detection was made with T=32, i.e. 32 line directions were possible in 11 1/4° increments, and R=17 since only 2 pixel wide lines within ± 16 pixels of the center of a 60 x 60 window were considered. Each detection made at (θ,r) in the coarse resolution process was refined as follows. A new set of $T \cdot R=7:5=35$ templates were established with 2° directional resolution and line width of 1 pixel. The parameter space tested was $\{\theta-6,\theta-4,\theta-2,\theta,\theta+2,\theta+4,\theta+6\} \times \{r-2,r-1,r,r+1,r+2\}$ where θ and r were the parameters gotten from coarse detection. Directional resolution finer than 2° would have required windows larger than 60 x 60 pixels.

The third enhancement to Hough line detection was rendered by checking for compatibility between the gradient direction at the high contrast point and the gradient direction of the candidate template for a line. Typically a high contrast point was considered to belong to at most 5 of the possible $T \cdot R$ lines. The actual number of possibilities was dependent on the resolution used in the gradient extraction. Thus, the use of gradient information further increased the speed of execution and at the same time sharpened the output of the detector.

2.2.2 Examples of Hough detection

Once the previously described procedure was implemented and its parameters were tuned to the imagery at hand, a great deal of testing was performed with no further changes made to the algorithm. In several cases extracted edges were used for registration or object detection experiments as described in Section 4 of this report.

Figure 11 shows an airfield image and two subimages taken from it. Coarse Hough detections made on the subimages are shown in the lower right. Each straight edge element shown is at most $60\sqrt{2}$ pixels long since 60 x 60 pixel windows were used to cover the images. Many good edge elements have not been detected as a result of the 5% point selection process employed. This effect is particularly prominent at the intersection of the three walks where widespread edge activity causes the highest contrast point set to be scattered and incapable of causing a strong response in any single template. In both images some edge elements overshoot their true length. This is because the responding templates are plotted rather than just the points inside them. In general further clean up is needed to delimit the true size of detected edge elements. In order to detect all nearly straight edges of length 30 pixels or more in the presence of noise templates 2-pixels wide were used and the detection threshold was set to 30 a priori. Since templates contained roughly 120 pixels roughly 1/4 of its points had to respond in order for the template to respond. Notice in the lower right of Figure 11 that high contrast points on the short side of one building triggered two templates in two different 60 x 60 windows.

Figure 12 shows a poor image of an airfield and the resulting coarse Hough detections. The directional resolution of the edges is unsatisfactory. The refinement procedure discussed in Section 2.2.1 produced the results shown in Figure 14. Excellent alignment of many of the refined edge elements allowed a simple procedure to combine short edge elements into long ones as shown in Figure 15. As will be shown in Section 4, the partial straight edge detection shown in Figures 14 and 15 is sufficient to establish registration with a map and thus unlock model information useful for focused reexamination of parts of the image. The poor quality GAFB image was deliberately chosen to illustrate this point. Before passing on it is important to note that the light/dark relationships along edge elements is indicated in Figures 14 and 15 while they are not evident in Figures 11 and 12. The dark side of the edge will be at the right as the edge is traversed in the direction of the arrow.

Figure 13 shows a large area with fine detail. Fine resolution Hough detections for this image are shown in Figure 16. Very little high level structure is evident in Figure 16 and perhaps more extraction effort should have been invested -- for example, in using more and smaller windows. However, the edge content shown in Figure 16 proved to be sufficient to register the image of Figure 13 with a map made from Figure 7. (The detected edge directions from Figure 13 were actually reversed to get Figure 16 because Figure 13 is a negative rather than positive as is Figure 7.)

2.3 Points of special curvature

It has been known for a long time that points of high curvature and inflection points on the boundary of an object contain most of the information used by humans in recognizing the object. Such points also play an important role in representation of an object in a compact form. A summary of a good deal of work in this area can be found in [Pavlidis 1977].

Figure 17 shows a tracing of a few major features from a 1:250,000 map of the Harrisburg, Pennsylvania region. These are features which should clearly be evident in aerial photography and perhaps even in LANDSAT imagery. There are several points whose uniqueness make them vital to recognition or registration of the region. Some of these points are intersection points, for instance, the juncture of the Pa. Turnpike and Route 15. Perhaps a dozen good points of high curvature exist. The crooked profile of Sherman Creek provides the greatest opportunity for recognition or registration - - 10 points of high curvature are available. The Juniata River contains interesting bends but the Susquehanna does not. There is, however, a sharp-cornered island down river from Harrisburg which has prominent features. A thin resevoir with 3 sharp corners is evident in the top right quadrant. As will be shown in Section 4, it is not necessary that all of the features of an image be recognized before the image itself can be recognized. It is also not necessary that continuous curves be extracted. For instance, segments of Sherman Creek are likely to be disconnected as the creek ducks under thick foilage. Thus, only some distinguishing features will be available in any given image of an area, but there should always be enough for recognition.

Besides the shape information from the curve in the neighborhood of a high curvature point, there might also be qualitative information, especially if the imagery is multispectral. For instance, the points on the small streams are linear water features inside a land/vegetation background. This water versus land quality can be picked up automatically from the multispectral signal. The corners of Hill Island would be defined by land/vegetation jutting out into open water. Boundary points so defined by local shape and region features could easily be extracted automatically with an acceptable degree of reliability and matched to a geographic data base for recognition and registration purposes.

Some experimentation was carried out in the detection of high curvature boundary points. The curve extraction procedure of Section 2.1 was applied to imagery to produce segments of boundaries. The curvature of each boundary segment was computed by a method similar to that in algorithm 7.1 of [Pavlidis 1977] and curve segments with points of high curvature were identified. All other boundary segments were discarded from this process. Figure 18 shows five "corners" identified in a window of the AFB image in Figure 11. The window contains the second airplane from the bottom and the wing of the first airplane. All three wing tips were extracted but the nose and one tail tip of the complete plane were missed. Some structured noise was also extracted. It should be clear that this evidence, along with other evidence such as straight edge content, is useful for recognizing objects and determining their position and the scale of the imagery. Further treatment follows in Section 4.

2.4 Other primitive image features

The requirements on primitive features are (1) that they be simply defined and (2) that they can be extracted automatically from imagery with acceptable reliability. In addition to the primitives mentioned in Sections 2.1-2.3 two others are presented. These primitives were, in fact, already briefly mentioned.

The intersections between two line or edge features can provide very good features for recognition. Due to the fact that edge detectors tend to be unstable at intersections some higher level (but still automatic and bottom-up) decision-making is required to extend detected boundary segments and force the intersection. It is even possible to create imaginary intersections as the surveyor does; for instance, to create the intersection of the wall of a building (extended) and a street. Intersections can create a local topology and geometry that provides reliable matching to a stored representation. Work has already been done in this area by [Zahn 1974] and [Dudani 1977]. Experiments with the use of simple intersection features is discussed in Section 4.

Boundary segments can be useful features even though the segment is not straight or of high curvature. The boundary may be significant due to the types of regions which it separates. This is particularly relevant if multispectral imagery is available to make region extraction and interpretation a lower level process. The fact that a boundary segment separates land and water regions does not give it unique properties for matching to a reference, especially if the segment shape is bland. However, the number of possible matches in a reference data base may be small, and global consideration of several such ambiguous features could yield a unique match between imagery and reference. A method for integrating ambiguous local matching evidence to form a unique global match is given in Section 4. Relaxation labeling is another technique for arriving at a global interpretation from ambiguous local interpretations [Zucker 1976, Tenenbaum 1976].

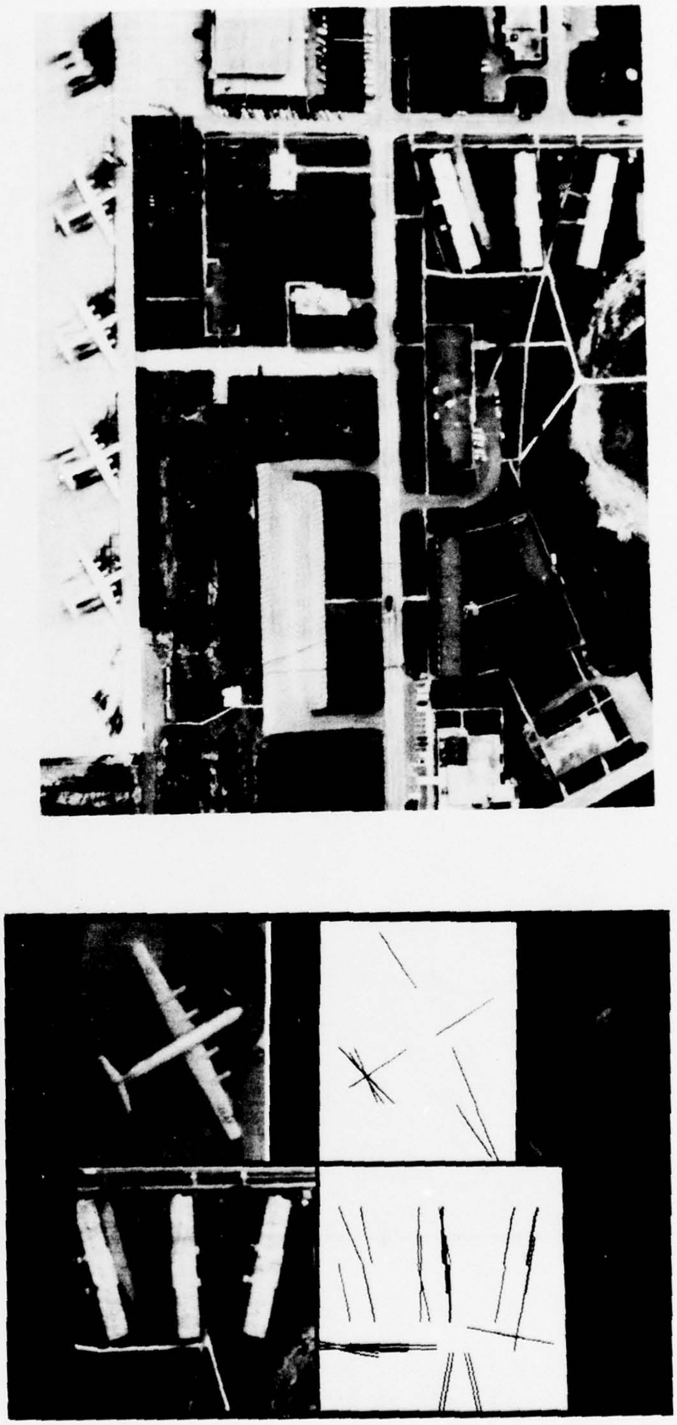


Figure 11. AFB test image and coarse Hough detections of straight edges in selected windows. AFB1 and AFB 2 (Image is about 2000 x 2000 6-bit pixels while windows are 250 x 250 and 500 x 500.)

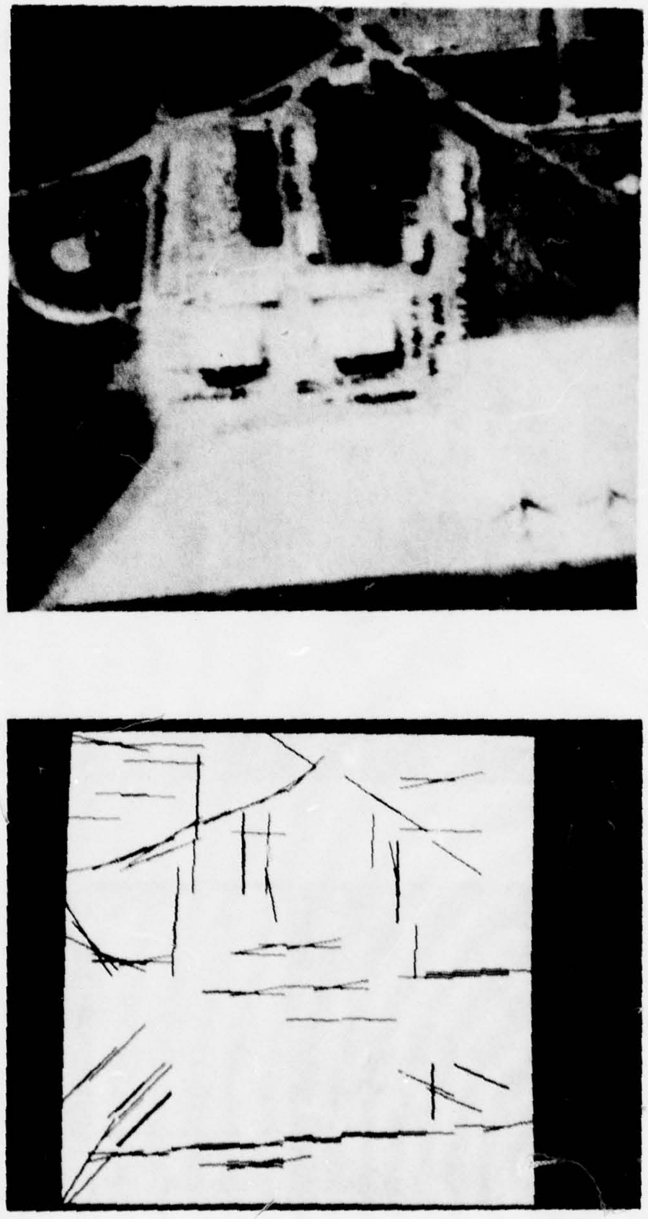


Figure 12. GAFB test image and coarse Hough detections of straight edges. (Image is 500 x 500 6-bit pixels.)

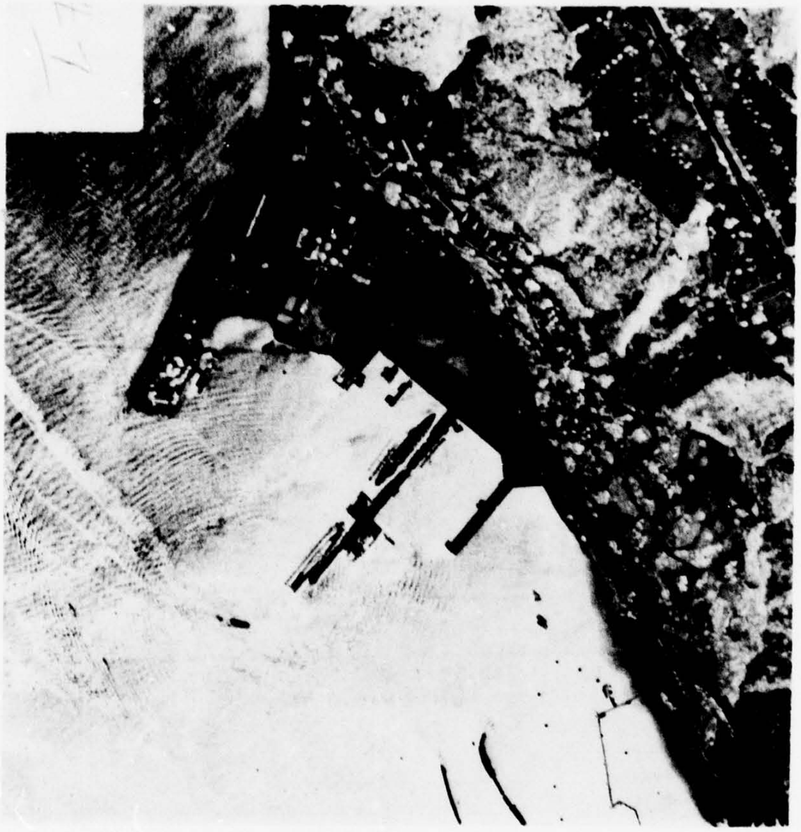


Figure 13. UNB1 test image: repeat coverage of naval base in Figure 7. (Image is about 2000 x 2000 6-bit pixels.)

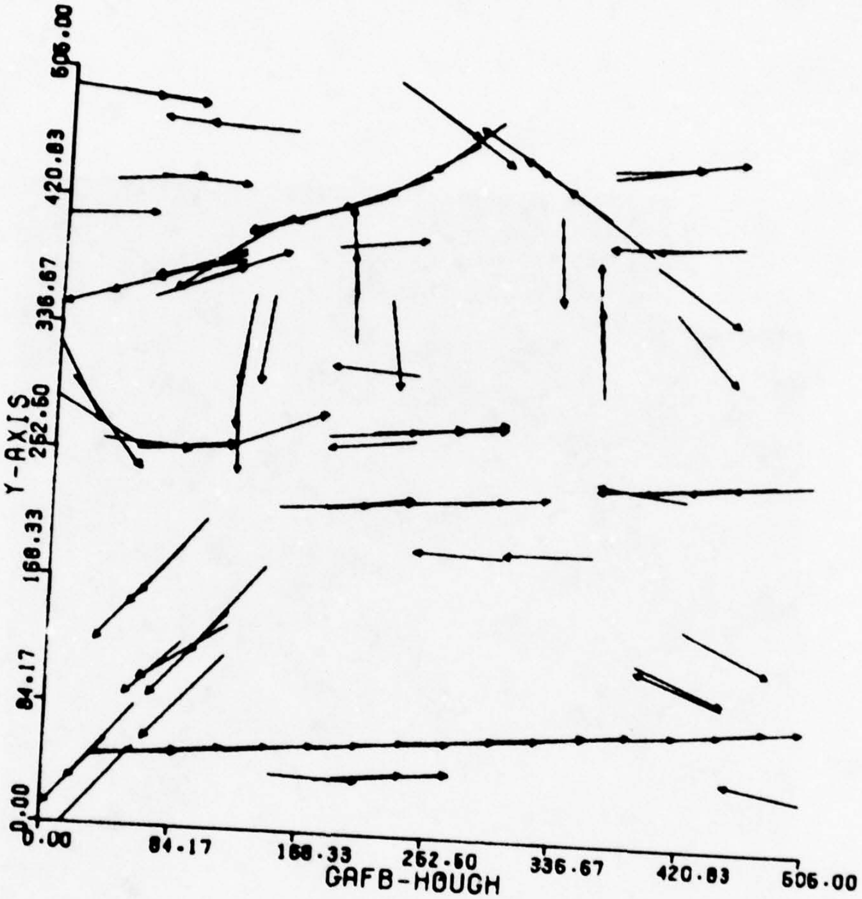


Figure 14. Coarse Hough detections from GAFB refined to 2° directional resolution.

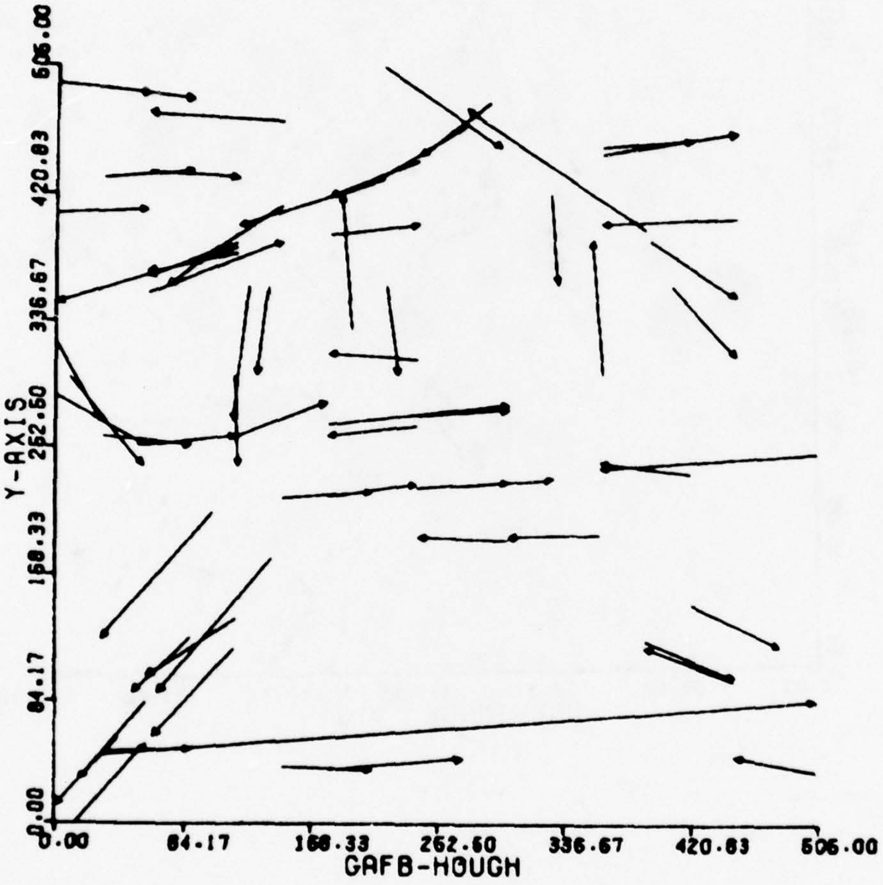


Figure 15. Hough detections from GAFB combined to form long non-overlapping straight edges.

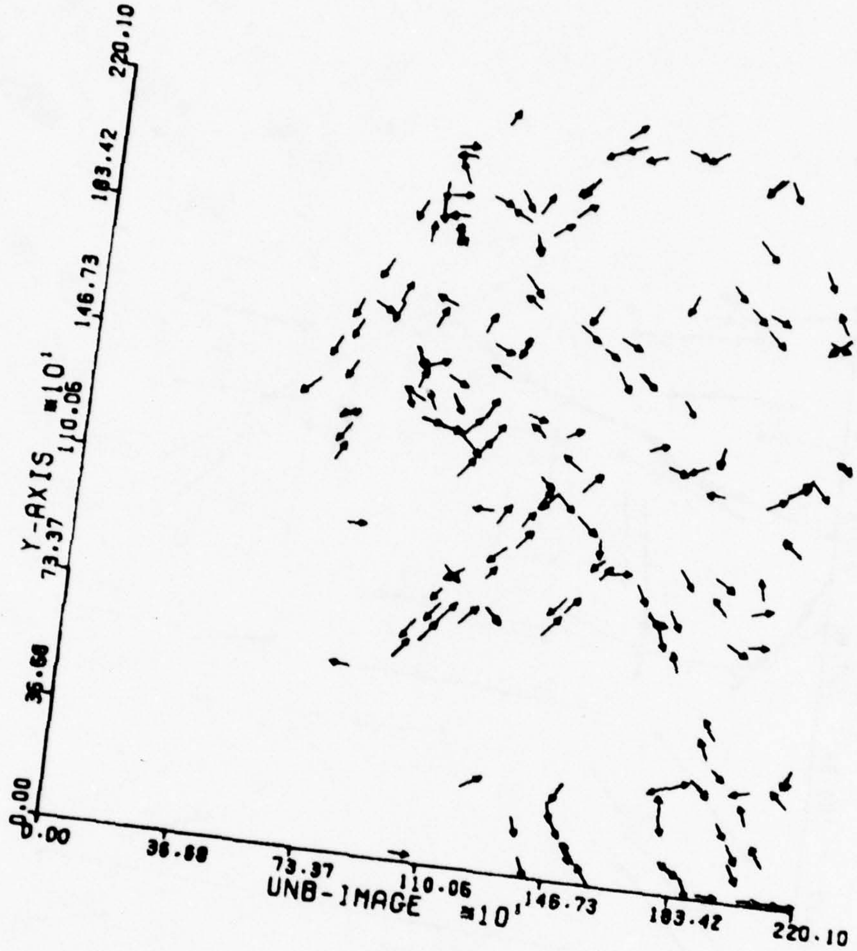


Figure 16. Hough detections from UNB1 image with 2° directional resolution. (Note: gradient direction was reversed due to negative image.)

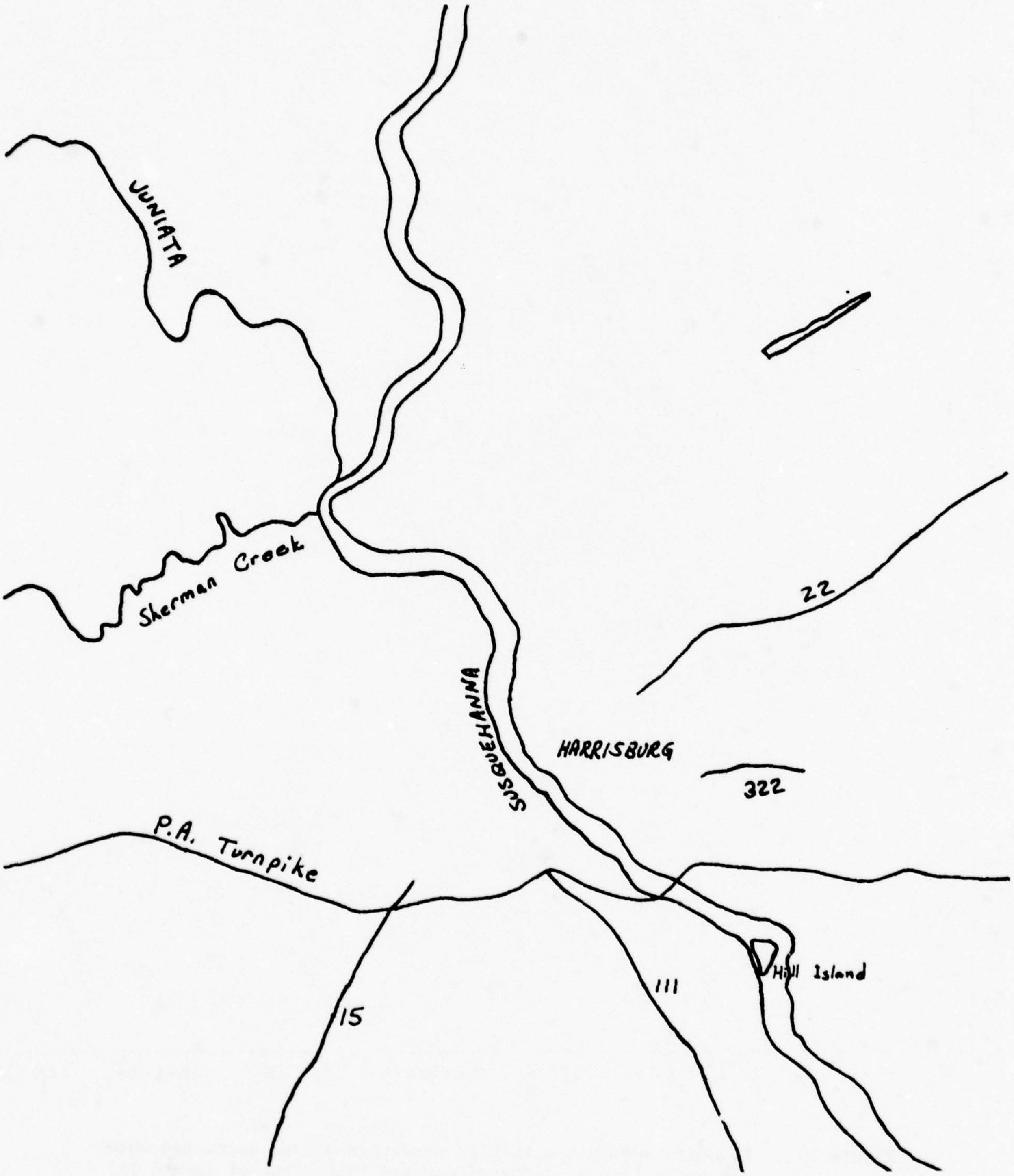


Figure 17. Prominent lineal cartographic features in the area of Harrisburg, Pennsylvania.

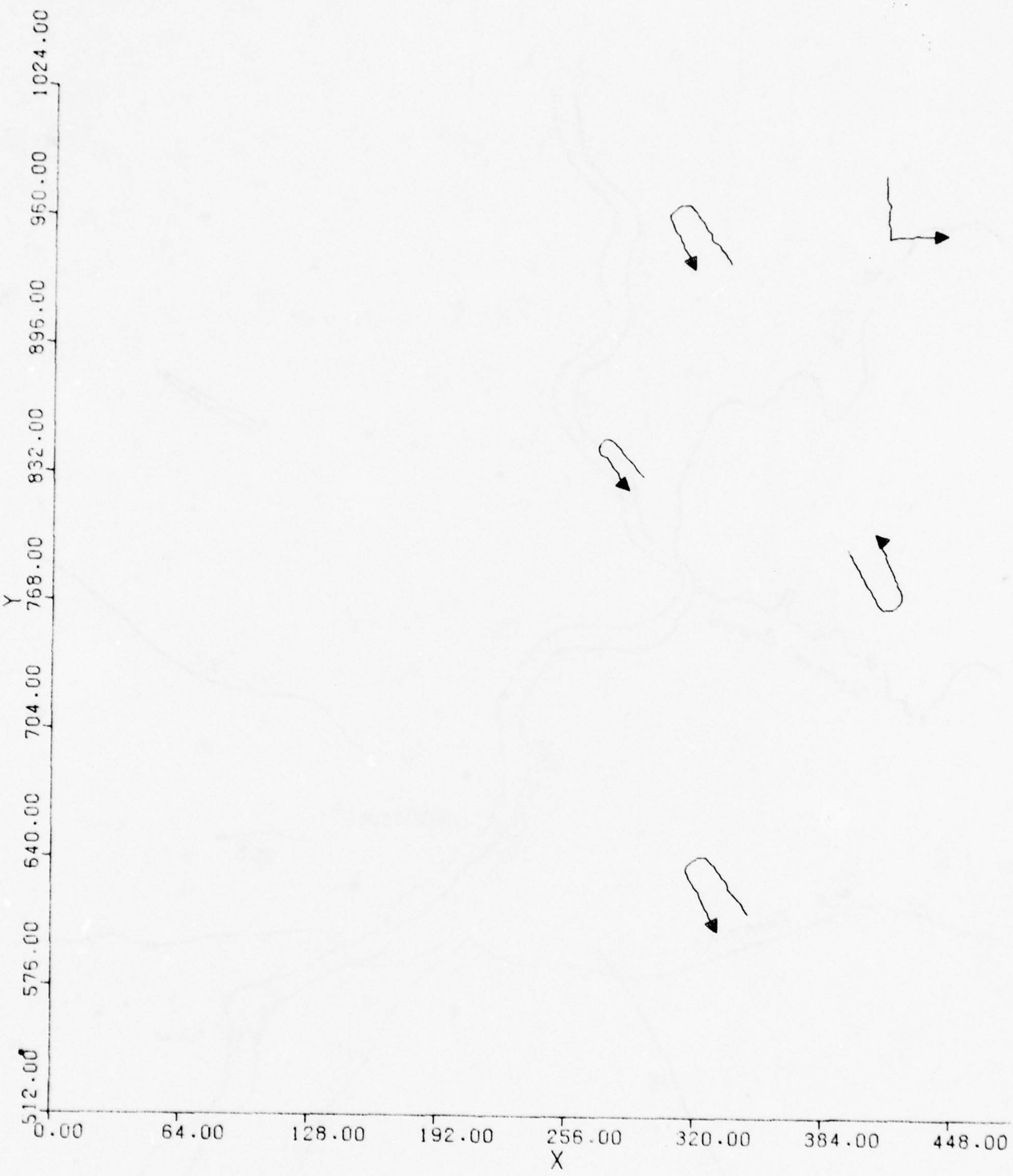


Figure 18. Boundary segments with high curvature points extracted automatically from a window of the AFB test image of Figure 11.

3. Recognition of structures via grammar models

Grammar models, originally introduced to model the structure of language, have maintained the interest of pattern recognition researchers for over a decade. Context free grammars in particular allow tractable hierarchical modeling of component structure. Initially conceived for linear strings, grammars have been generalized to apply to 2-D as well, either by changing the grammar model itself [Shaw 1970] or by analyzing only 1-D boundary curves in an image [Ledley 1966].

3.1 Background and motivation of a grammatical approach

There have been many efforts in linguistic pattern recognition. The work of Shaw [1970], Pavlidis [1977], and Fu [1974] are exemplary and contain much discussion of the virtues of the linguistic approach. Many weaknesses of former linguistic pattern recognition implementations stem from the fact that pattern recognition researchers did little to tailor linguistic analysis methods to the more demanding real data situation. First of all, most implementations commit themselves to unique segmentations in "preprocessing" stages which do not utilize available structural knowledge and thus irrevocable decisions are made in locally ambiguous contexts. Secondly, implementations have been one directional. Analysis is either done in a bottom-up (data-directed) or top-down (model-directed) fashion but not in both directions. Shaw's PDL analysis scheme was top-down with primitive processing always done under model hypothesis. The approaches of Pavlidis and Fu are characteristically bottom-up with early commitment to unique segmentations in the absence of structural hypotheses.

It is the author's claim that bottom-up techniques which segment without model knowledge cannot succeed in complex data environments. Take for example the analysis of the edge structure in a reconnaissance image. Using global parameterization the edge detector is doomed to pass either too many edges or two few. Too many edges may fool the structural analyzer or will at least overwork it. Too few edges may cause analysis to fail. A case is the detection of a rectangular building. Due to sun angle possibly only 2 or 3 sides of the building will be picked up by an edge detector using sensible thresholding. Any thresholding that would detect the weak sides would necessarily detect an enormous number of edges in other textured regions of the image. What is really wanted is an analysis technique that would search for the weak sides locally under model control. (Griffith [1973] addresses this strategy and analyzes it as a work saving device rather than a device necessary for accurate recognition.) On the other hand, purely top-down analysis strategies are impractical in complex problems because too much work must be expended in the formation of hypotheses which are consistent with the model but in no way relate to the untested data at hand.

A possible solution is proposed below. Non-directional analysis can be achieved by identifying reliably extracted primitive components of the model, extracting those in preprocessing without structural constraint and then doing model-directed search for remaining pattern structure. Thus analysis can proceed in either the bottom-up or top-down direction. The dichotomy of terminal and nonterminal structures is retained here - - terminal structures are recognized only through primitive feature extraction on real imagery while nonterminal structures are processed only in the higher level syntactic/semantic model space. However, unlike most other approaches, recognition of terminal and nonterminal structure is overlapped in time with the data processing and model processing providing each other with guiding feedback. Section 3.2 outlines the theoretical development of such a non-directional analysis. Section 3.3 describes a simple example of non-directional analysis in the detection of rectangles in reconnaissance imagery. Primitive extraction is treated in more detail in Section 3.4 and a final discussion of issues follows in Section 3.6.

3.2 Outline of a theory for non-directional structural pattern recognition

Due to space limitations complete definitions, proofs, and discussion of concepts cannot be included here. Instead, certain basic background is assumed and only a broad treatment is given. Excellent informal treatment of problem reduction representations (PRR), also known as AND/OR graphs, and state space representations (SSR) can be found in the text [Nilsson 1971]. Formal treatment can be found in a paper [VanderBrug and Minker 1975] and in a dissertation [Stockman 1977]. Also relevant is a paper by Hall [1973] showing the equivalence of a context free grammer (CFG) to a finite AND/OR graph, and a paper by Chang and Slagle [1971] showing that conversion can be made from PRR to SSR so that the A* algorithm can be used to produce solutions of AND/OR graphs. The practical result of integrating this work is as follows. Structural constraints on real-world objects can be modeled by a CFG or its equivalent PRR. Recognition of the object then amounts to parsing data using the PRR. Recognition results in a parse tree (CFG) or a solution tree (PRR) which is a hierarchical breakdown of each object structure in terms of its components recognized in the data.

In order to effect an efficient non-directional analysis special embellishments are appended to the usual PRR. First of all, AND successors are ordered and are searched for sequentially and only after all previous successors of the set are solved. For example if problem A is solved by solving both problems B and C, only one of the subproblems B or C will be posed at a time. There is no sense wasting effort to solve B if C is unsolvable. This strategy was used in a top-down parser by Chartres and Florentin [1968]. The *first* AND successor of a set of subproblems of problem P is called a *primary successor* of problem P. Every OR sucessor of problem P is called a primary successor. A *primary descendant* of the root problem R is either a primary successor of R or the primary successor of some primary descendant of R. In the linguistic pattern recognition context, primary terminals are key primitives or prominent features which can be reliably detected without syntactic constraint. Recognition of a primary problem would then trigger the search for the solution to problems which have the solved problem as primary successor. Search for this solution would typically involve a top-down search for the solution of other non-primary successors. If the inverse of the primary successor relation is available in the PRR, analysis can proceed recursively in either bottom-up or top-down direction. CFG's (hence finite AND/OR graphs) are easily inverted for bottom-up analysis. If $A \to BC$ (problem A is solved by solving both problems C and B) and $D \to EBG$ then goals A and D should be initiated if a solution to structure B were at hand. Separate (parallel) model-directed searches would then be done for the solution of successor C of A and the higher priority of successors E and G of D.

In [Stockman 1977] a conversion is made from PRR to SSR which has the following properties.

(1) PRR has a solution graph if and only if SSR has a solution path.

(2) All solutions to PRR can be found in a top-down mode by search of SSR with the initial state encoding the root problem of PRR.

(3) All solutions to PRR can be found in a bottom-up-top-down mode by search of SSR with a set of initial states, each one encoding some solved primary descendant of the root problem.

Any of the standard search algorithms of SSR [see Nilsson 1971] will do--depth-first, breadth-first, or ordered search. In applications discussed below a heuristic function evaluating the merit of partial solutions was used which enabled A* search. Note that bottom-up initiation of search (point 3 above) requires that PRR have a finite set of primary primitives, which is the case with a CFG.

3.3 An experiment in the recognition of rectangles

In this section a simple, but non-trivial example is given of the non-directional analysis algorithm outlined in Section 3.2. Actual computer runs on real and simulated data have been made and have demonstrated the capabilities of the analysis paradigm. The non-directional analysis algorithm was first implemented as the structural component of a waveform parsing system [Stockman 1977] and was rigorously studied in the recognition and measurement of pulse waves. The identical structural component was then applied to the recognition of rectangular objects in images as described below. The transition from 1-D to 2-D data was enabled by the system's treatment of locational information as attributes of structures.

Problem specific procedural semantics were necessary to handle attribute manipulation in each application and were coupled to the structural analysis in a uniform way.

3.3.1 The experimental data

Figure 19 shows the simulated experimental data. Input to the recognition system is a set of undirected edge elements each specified by two points. The data is rough because no complete contours exist and there are gaps and changes in orientation along the sides. Complex corners could fool ordinary tracking algorithms. This data, however, is probably better than can be expected from preprocessors in many applications. Generally it should not be expected that edges sufficiently characterizing object structure can be delivered by model-independent preprocessing. Suppose, for instance, that edge element 9 was quite faint in the image. Globally parameterized edge detectors would then not deliver that edge element. There is, however, a solution to this problem in model-directed local edge detection. Suppose that the sides $\overline{DA}$, $\overline{AB}$, and $\overline{BC}$ of rectangle ABCD were recognized at a certain point in the analysis. At that point edge CD could be hypothesized and the image scanned under lenient parameters. The data of Figure 19 should therefore be regarded as a union of two sets of edges, those primary edge structures detected under stringent global parameterization and those secondary edge structures detected locally under lenient parameterization. In the actual computer runs edge elements #7 and #12 were used as primary edge structures, but arbitrary choices could have been made for starting the search.

3.3.2 The rectangle model

Figure 20 shows the PRR graph model of the rectangle domain. STRAIT is the node activated by primary detection of a long strong straight edge element. Recognition of STRAIT causes recognition of the first side of a rectangle <PRIM>. This first side may also be extended in the backward or forward direction (BACKEX, FORWEX) to get other starting sides to explore. Each starting side <PRIM> recognized activates the search for <LFRC> and <RGRC> which are complete rectangles formed by respectively counter-clockwise and clockwise accumulation of the other 3 sides. The <SIDE> structure and all its substructures are secondary, that is, their recognition depends upon detection of edges in precise relation to those recognized previously along the path. The coupled semantic routines handle the angular structure and check for approximately equal opposite sides. A context-free grammar corresponding to the PRR of Figure 20 is given in Figure 21. It is important to emphasize the role of semantics in restricting the search for detections and the recognition of higher level structures. For instance, each time the <SID1> structure has been recognized but its length is out of range that analysis path is terminated. Similarly if the length of the <SIDE> structure does not match a parallel side the analysis path is terminated.

3.3.3 Quality of substructures and merit of states

Each substructure of the PRR must be assigned a quality $o \leq \alpha \leq 1$ when recognized to reflect the confidence (fuzziness?) of recognition. For primary primitive structures this quality is gotten from the detector itself, but cannot depend upon structural context. For secondary primitive recognition, the recognition quality depends

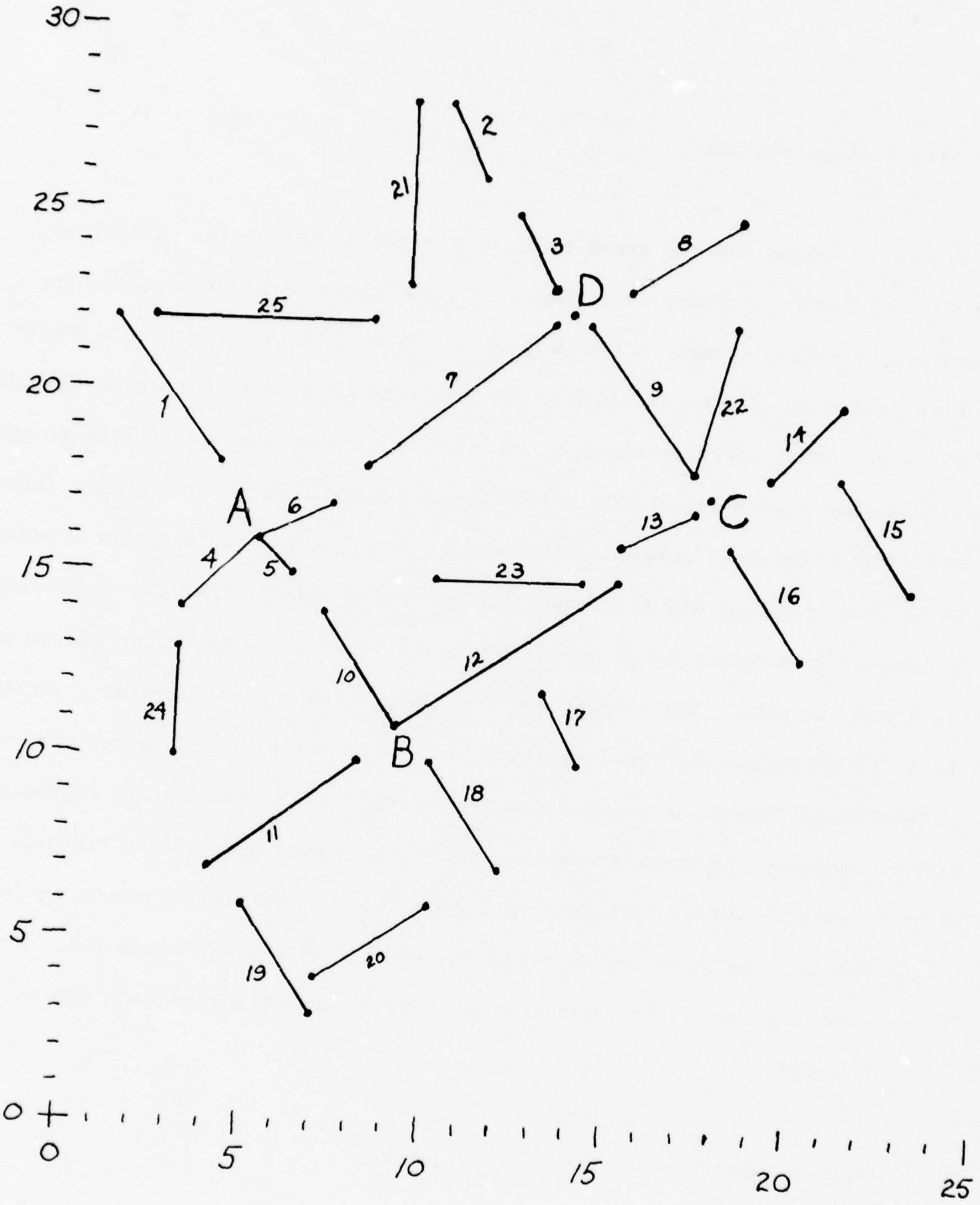


Figure 19. Simulated data set for rectangle recognition experiment.

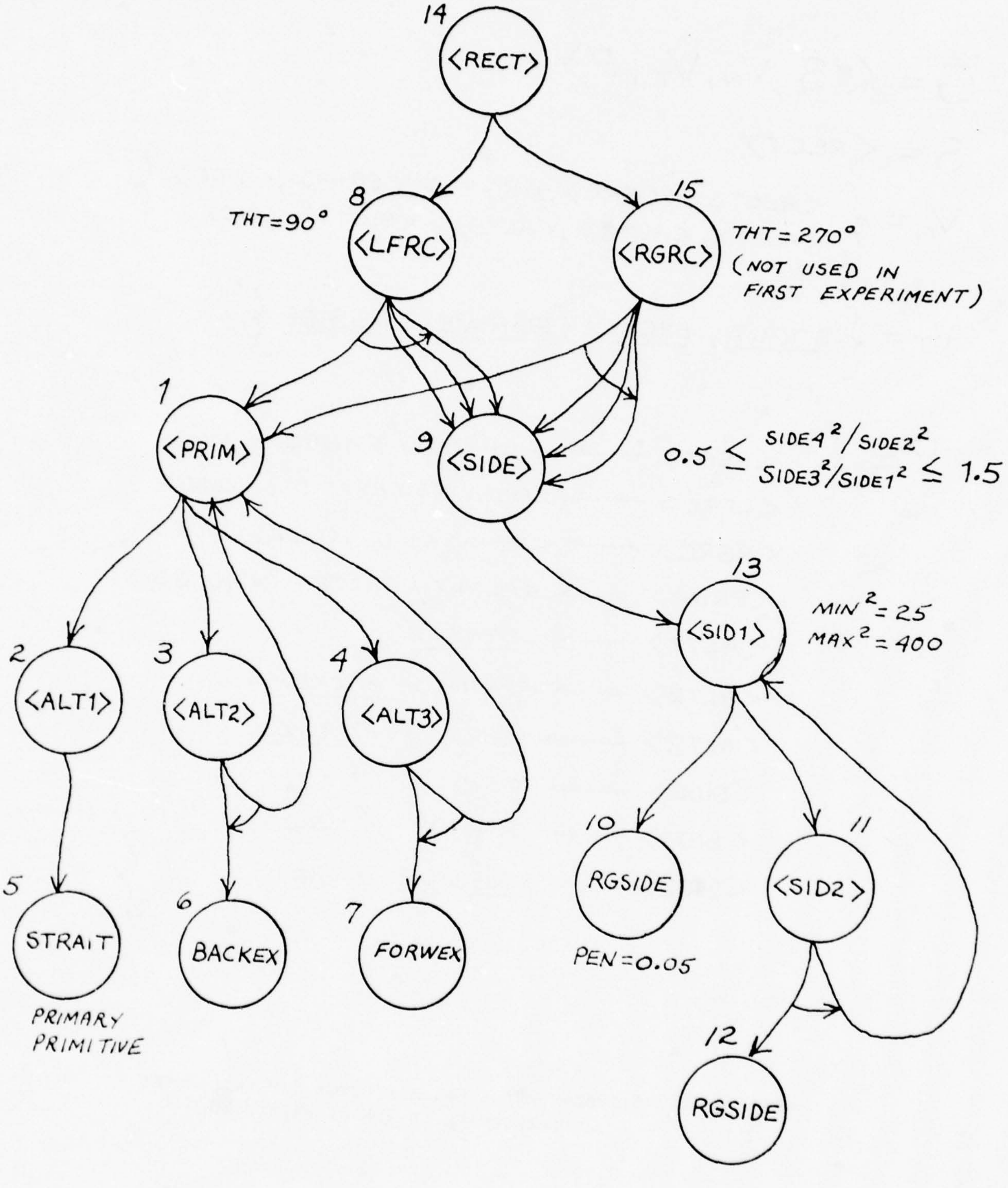


Figure 20. Problem Reduction Representation (PRR) for rectangle recognition.

$$G = \langle S, V_N, V_T, P \rangle$$

$$S = \langle \text{RECT} \rangle$$

$$V_N = \left\{ \begin{array}{l} \langle \text{RECT} \rangle, \langle \text{LFRC} \rangle, \langle \text{RGRC} \rangle, \langle \text{PRIM} \rangle, \langle \text{SIDE} \rangle, \\ \langle \text{ALT1} \rangle, \langle \text{ALT2} \rangle, \langle \text{ALT3} \rangle, \langle \text{SID1} \rangle, \langle \text{SID2} \rangle \end{array} \right\}$$

$$V_T = \left\{ \underline{\text{STRAIT}}, \underline{\text{BACKEX}}, \underline{\text{FORWEX}}, \underline{\text{RGSIDE}} \right\}$$

$$P = \left\{ \begin{array}{l} \langle \text{RECT} \rangle \longrightarrow \langle \text{LFRC} \rangle / \langle \text{RGRC} \rangle \\ \langle \text{LFRC} \rangle \longrightarrow \langle \text{PRIM} \rangle \langle \text{SIDE} \rangle \langle \text{SIDE} \rangle \langle \text{SIDE} \rangle \\ \langle \text{RGRC} \rangle \longrightarrow \langle \text{PRIM} \rangle \langle \text{SIDE} \rangle \langle \text{SIDE} \rangle \langle \text{SIDE} \rangle \\ \langle \text{PRIM} \rangle \longrightarrow \langle \text{ALT1} \rangle / \langle \text{ALT2} \rangle / \langle \text{ALT3} \rangle \\ \langle \text{ALT1} \rangle \longrightarrow \underline{\text{STRAIT}} \\ \langle \text{ALT2} \rangle \longrightarrow \langle \text{PRIM} \rangle \; \underline{\text{BACKEX}} \\ \langle \text{ALT3} \rangle \longrightarrow \langle \text{PRIM} \rangle \; \underline{\text{FORWEX}} \\ \langle \text{SIDE} \rangle \longrightarrow \langle \text{SID1} \rangle \\ \langle \text{SID1} \rangle \longrightarrow \underline{\text{RGSIDE}} / \langle \text{SID2} \rangle \\ \langle \text{SID2} \rangle \longrightarrow \underline{\text{RGSIDE}} \; \langle \text{SID1} \rangle \end{array} \right\}$$

Figure 21. A Context Free Grammar (CFG) for rectangles corresponding to PRR of Figure 20.

upon the quality of detection and upon the degree to which the detection satisfies the structural hypothesis. The quality of a non-primitive structure is defined as the minimum quality of its substructures. This definition might not create the "best" recognition procedure but it does create an admissible search for the best interpretation. The merit of a path in model space is defined as the minimum quality of any structure recognized along that path. The ordered search for interpretations will thus find the highest quality one first because it always extends the highest merit path first.

3.3.4 An example of processing

The non-directional algorithm was started on the data of Figure 19 with the syntactic binding STRAIT = (13,22) - (8,18), that is, the primary terminal of the grammar was identified to be the straight edge element directed from point (13,22) to point (8,18). Significant states of the resulting state space search are described below. Each state is a partial parse tree and has a merit computed from the recognized terminal structures in it. By state #3 the <PRIM> structure is recognized and the grammar immediately causes three states to be generated, one each to search for <ALT2>, <ALT3>, and <LFRC> respectively. The <ALT2> alternative attempts to extend the side backward while the <ALT3> alternative attempts a forward extension. The <LFRC> alternative sets the goal of finding a second side at a 90° bearing from the first. In states 4 to 10 the <LFRC> alternative is pursued but no such perpendicular side exists and the search path deadens. <ALT3> does succeed in a forward extension of <PRIM> to point (5,16). In so doing, the merit of states on this path drops from 1.0 to 0.9. This enables <ALT2> to be pursued in states 13 and 14 which produce no backward extension. The new <PRIM> structure recognized from point (13,22) to (5,16) once again causes 3 alternative search goals to be set--forward, backward, and perpendicular extension. States 19 to 48 pursue a perpendicular extension from point (5,16) to (12,7), but point (12,7) is a dead end since no further extension is possible. At state 53 another open path is picked up and a perpendicular extension is driven from point (9,11) to point (17,17). Thus by state 72 three sides of rectangle DABC are recognized. By state 97 a path is driven perpendicular to side BC to point (10,28) thus completing the recognition of 4 perpendicular sides. However, the path overshoots the correct beginning point of the rectangle and becomes dead due to a semantic check on the sizes of sides 2 and 4. An alternate open path is pursued to final state 106 causing recognition of rectangle DABC. Two open paths remain by state 106 representing paths D to (3,14) and D to (5,16) to (9,11) to (15,15) respectively but can not develop into recognition of other rectangles.

A more detailed presentation of the search is now given. At any state of the search there may be one or more partial matches of the model (Figure 20) with the data (Figure 19). Each partial match is rated for its quality and this rating is used to determine which analysis is extended in the next search state. The example search was started with state #1 rated as 1.0 and encoded as follows.

```
     5            5
#.   [ 13,22,8,18]
     1            1
```

The meaning of this encoding is that structure 5, i.e. the STRAIT structure in the rectangle model of Figure 20, has been recognized spanning points (13,22) and (8,18) of the image. Structure 5 is substructure 1 of its parent structure. The dot "." to the left of the bracket "5" indicates that processing is focused on structure 5.

Using the model the search algorithm recognizes that structure 2 exists and generates state #2 encoded as

```
        2            5             5 2
   # .  [ 13,22,8,18[ 13,22,8,18] ]
        1            1             1 1
```

and also rated at 1.0.

Recognition of structure 2 implies recognition of structure 1 at state #3 of the analysis. Special processing indicated in the model (but not indicated in Figure 20) causes the state encoding to be collapsed into

```
        1             1
   # .  [ 13,22,8,18  ]  .
        1             1
```

The model indicates that the < PRIM > structure 1 can be part of 3 different superstructures. Thus at state #4 of the analysis there are three competing parallel interpretations encoded as follows.

```
     8     1              1 8
   # ( .  [ 13,22,8,18  ] )
     1     1              1 1

     4     1              1 4
   # ( .  [ 13,22,8,18  ] )
     1     1              1 1

     3     1              1 3
   # ( .  [ 13,22,8,18  ] )
     1     1              1 1
```

The first interpretation has structure 1 as the complete first side of the rectangle. The second and third alternatives see structure 1 as an incomplete side that must be extended in the forward or backward direction. All three alternatives are rated 1.0, the top one is taken for expansion next in the search. Using the model, the search generates the following encodings in a top-down manner. Note the 90° direction change as specified in the model for searching for side 2 with respect to side 1.

```
  8   1            1 8
# ( . [ 13,22,8,18 ] )
  1   1            1 1
```

```
  8   9            9 1            1 8
# ( . ( 8,18,11,12 ) [ 13,22,8,18 ] )
  1   2            2 1            1 1
```

```
  8 9              13          13 9 1            1 8
# ( ( 8,18,11,12 . ( 8,18,11,12 ) ) [ 13,22,8,18 ] )
  1 2              1            1 2 1            1 1
```

```
  8 9            13             10         10 13 9 1            1 8
# ( ( 8,18,11,12 ( 8,18,11,12 . ( 8,18,11.12 ) ) ) [ 13,22,8,18 ] )
  1 2            1              1            1 1 2 1            1 1
```

DEAD

This line of analysis deadens because structure 10 is a primitive straight line structure for which there is no above threshold evidence in the data. An alternate course of analysis is thus pursued as follows.

```
  4   1            1 4
# ( . [ 13,22,8,18 ] )  rated 1.0
  1   1            1 1
```

```
  4              7           7 1            1 4
# ( 13,22,8,18 . ( 8,18,3,14 ) [ 13,22,8,18 ] )  rated 1.0
  1              2           2 1            1 1
```

```
  4              7           7 1            1 4
# ( 13,22,8,18 . [ 7,17,5,16 ] [ 13,22,8,18 ] )  rated 0.9
  1              2           2 1            1 1
```

The < PRIM > structure has been extended forward to point (5,16) but at the expense of shooting a gap : hence the rating is reduced in proportion to the gap size to 0.9. Structural alternative < ALT2 > is pursued temporarily because of higher rating 1.0 but after failure the lineof analysis just above is again taken up as the highest rated alternative. A few of the encodings along the path to a correct recognition are as follows.

AFTER RECOGNITION OF TWO SIDES $\overline{DA}$ AND $\overline{AB}$

```
  8             9           9 1            1 8
# ( 5,16,9,11 . [ 5,16,9,11 ] [ 13,22,5,16 ] )  rated 0.9
  1             2           2 1            1 1
```

AFTER RECOGNITION OF THREE SIDES $\overline{DA}$, $\overline{AB}$, AND $\overline{BC}$

```
  8              9            9 9           9 1            1 8
# ( 9,11,17,17 . [ 9,11,17,17 ] [ 5,16,9,11 ] [ 13,22,5,16 ] )  rated 0.9
  1              3            3 2           2 1            1 1
```

AFTER PROPOSING SEARCH FOR FOURTH SIDE $\overline{CD}$.

```
  8              9             9 9            9 9           9 1            1 8
# ( 9,11,17,17 . ( 17,17,11,25 ) [ 9,11,17,17 ] [ 5,16,9,11 ] [ 13,22,5,16 ] )  rated 0.9
  1              4             4 3            3 2           2 1            1 1
```

AFTER RECOGNIZING ENTIRE RECTANGLE

```
    8             9             9 9            9 9           9 1            1 8
# . [ 17,18,14,22 [ 17,18,14,22 ] [ 9,11,17,17 ] [ 5,16,9,11 ] [ 13,22,5,16 ] )  rated 0.9
    1             4             4 3            3 2           2 1            1 1
```

The automata that manipulates such encodings to perform the analysis is detailed in [Stockman 1977]. The manipulation of the search areas, or intervals was done in associated "semantic" routines. This was necessary so that the overall problem solving mechanism would work uniformly on waveform and image data.

3.3.5 Discussion of rectangle recognition experiments

The <RECT> goal in the rectangle PRR was not actually used as the root problem in the experiments. <LFRC> was used instead to control only a counter clockwise search. <RGRC> controls clockwise search and was not used to save time. It was the intention that in future work the confidence measure would be allowed to increase, and thus, while a path may block in one direction due to noise or distortion it may be found in the reverse direction after enough confidence has been built to overcome the noise.

A simple primitive detection module was programmed so that edge elements as pictured in Figure 19 could be extracted from grey scale images. The detector was used to verify the existence of an edge element as predicted by the grammar. The primary edge element STRAIT had to be recognized by other means. The detector scanned across a hypothetical edge and recorded points of maximum gradient magnitude. These points were then fit with a straight line to assign a confidence value to the hypothesis. The system was then tried on two of the rectangular buildings in the GAFB image. The searches were successful and were less bushy, i.e. more efficient, than those on the constructed example of Figure 19. However, too much adjustment was necessary to make the process work and little generality can be claimed.

3.4 More on the detection of shape features

Section 2 of this report discussed the detection of primitive features in imagery without use of a priori model information. Straight edge elements, smooth curve segments, and points of high curvature on them were discussed as useful features. It was argued that many image features could be detected by cheap implementations although a fair portion of existing features might be missed. Once enough image features are assembled hypotheses about the remaining image content can be raised and tested by using models such as PRR. Verification of a hypothesis is much more efficient than searching for a primary primitive without model guidance and thus more refined shape features can be afforded.

Verification of the presence of a boundary segment of a particular shape can be done using curve fitting, Hough detection, or searching for points of a prototype set of points. These techniques are discussed in some detail in Section 5. An experiment in the recognition of the curved tip of an airplane wing is described briefly here. It was easy to add a parabolic curve-fitter to the existing straight line fitter used in the rectangle experiment. As before, a specially shaped boundary curve was hypothesized in a given region of the image with a given orientation. Profiles were searched aligned with the hypothesized coordinate system and the high gradient points were fit with a parabolic curve. Goodness of fit determined the confidence in detection.

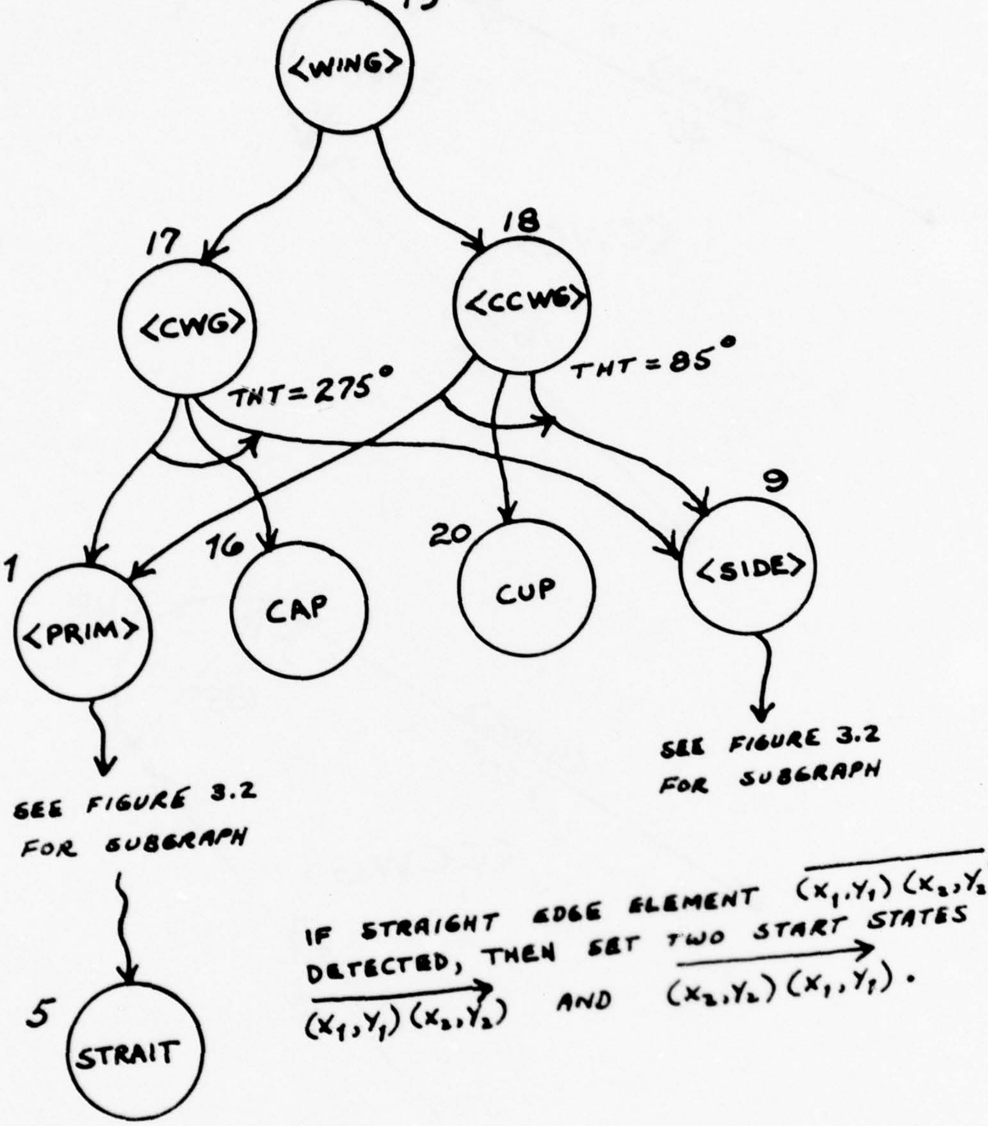
Figure 22 gives a PRR model for the wing of an airplane of the type found in the AFB1 image of Figure 3. Figure 23 shows the geometric structure modeled in the PRR. (Curvature constraints for the CAP and CUP and length constraints for the sides are omitted from Figure 22). The curvature parameter of the parobolic fit will be positive or negative as the wing boundary of 3.5 is traversed clockwise or counter clockwise respectively.

Search using the airplane wing PRR was never correctly carried out. The major problem was that 1-D fitting techniques were used. The fitter was confused by the many points existing at the juncture of the straight edge and the parabolic tip. These points had the same x value but different y values: the profile scanning could only select one point for each x value. 2-D curve fitting was clearly indicated but the experimentation was halted in order to pursue more promising avenues as discussed in Section 4.

3.5 Discussion of the use of grammar models

Shaw [1970] realized the value of using grammar modules for 2-D analysis. He was also aware of the difficulty in doing segmentation as preprocessing. The work reported here attempted to extend Shaw's work in two directions. First of all, bottom-up operators were employed and secondly, multiple competing parses were developed.

Martelli [1976] exploited the search concept. His approach is to dynamically search image data for continuing edge elements. A priori shape information was encoded in heuristic functions. Martelli has even experimented on the same rectangle detection problem discussed in this paper. No structural hierarchy was used and all



IF STRAIGHT EDGE ELEMENT $\overline{(x_1,y_1)(x_2,y_2)}$ DETECTED, THEN SET TWO START STATES $\overrightarrow{(x_1,y_1)(x_2,y_2)}$ AND $\overrightarrow{(x_2,y_2)(x_1,y_1)}$.

Figure 22. Problem reduction representation for recognition of an airplane wing.

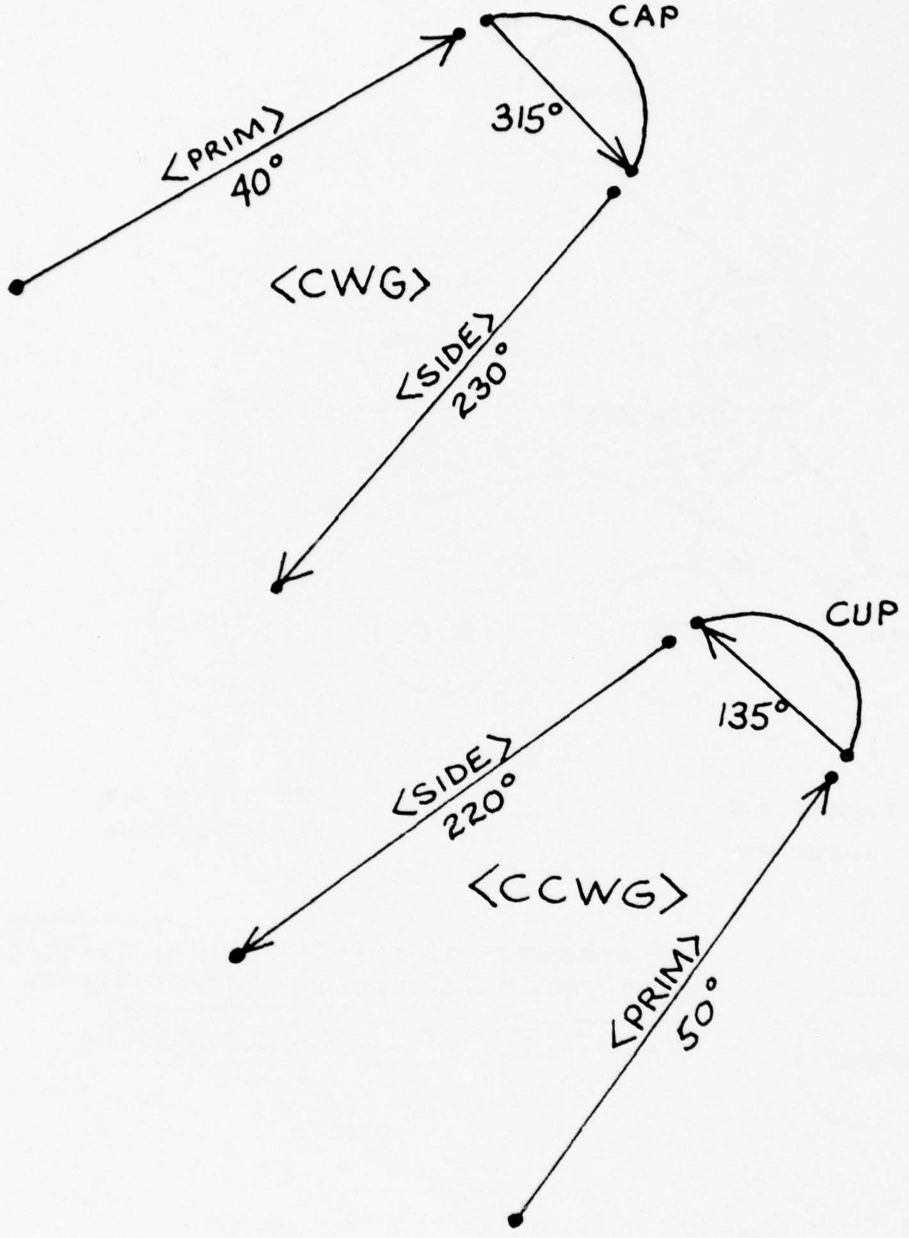


Figure 23. Geometric structure of an airplane wing modeled by PRR of Figure 22. (Arrows indicate order of edge traversal and not light/dark relation.)

operations of the search were in terms of single pixel extensions of the path. This study generalized the work of Martelli by using larger-than-pixel primitives and by using paths of a grammar to remember and control the shape of search paths in the data.

Miller [1973] used both bottom-up and top-down operations in a small speech recognition system, coining the term "island of reliability" for what are called "primary primitives" in this report. Griffith [1973] also did bottom-up followed by by top-down processing. Neither Miller nor Griffith, however, presented a general algorithm for carrying out such processing. This was the essential contribution of [Stockman 1977] and the topic of this section.

Few would argue that vast amounts of a priori knowledge are not necessary for real world scene analysis. By constructing a PRR certain a priori knolwedge is available for a uniform recognition process. Semantics coupled to the nodes of the PRR can help. But a finite PRR is equivalent to a CFG and so it is well known that we have a tractable but limited tool. Further work is necessary to test the viability of this non-directional analysis technique on complex recognition problems. Items learned from the current research follow.

The 1-D curve fitting technique used in this research and to be detailed in Section 5 should be replaced with a 2-D technique. If 2-D curve fitting is too expensive or unreliable then Hough curve detection should be tried. In general, quicker techniques with fewer parameters should be used in verifying hypotheses: most of the recognition information is contained in the structural constraints. Constraints other than those on shape should be incorporated into the PRR; for instance, a given region could be tested for a hot spot assuming that an IR image is also available, or a given region could be tested for color if color imagery were being used. PRR's that model loosely related composites should be tried. For instance we might have <AIRFIELD>--> <BLDG GROUP> <PLANE GROUP> < RUNWAYS> where an airfield is recognized as a composite of planes, buildings, and runways. More thought has to be spent in assigning the confidence value to < AIRFIELD> given the confidence values assigned to the components.

4. Iconic shape models

This section examines the matching of arbitrary shapes extracted from an image with shapes stored in a reference data base (RDB). There was considerable discussion of geometric shape features in Section 2. The term "iconic" is used to indicate that the image structure "looks like" the structure stored in the RDB. Mathematically we might define that structure A looks like structure B if there exists an RS&T transformation* mapping points of A onto points of B. That is in fact the formalism that is used in this section. Some lenience must be allowed in the paradigm so that (1) some rubber sheet distortion is permitted and (2) missing or additional parts are permitted in the structured image points. The first facility can be provided by use of approximate RS&T mappings which tolerate some distortion and the second facility can be provided by use of a tolerant partial matching procedure.

There are two specific applications to which this discussion is oriented. First of all we want to be able to register aerial imagery with an existing geographic (cartographic) data base (GDB). To do so we need to match image features with their icons stored in the GDB. The result of a correct global matching of each image feature to a corresponding GDB feature is an RS&T transform which registers all points of the image to the GDB coordinate system. As a result of the registration, the complete image can be examined for content in comparison to the iconic feature content stored in the GDB. Updates to the GDB can be made. For instance, rivers can be

*Rotation, Scaling, and Translation

searched in the image for the appearance of new bridges or airfields could be searched for the disappearance of planes. In a second case we want to be able to recognize generic moveable objects by matching image edge structure to iconic edge structure stored in a model (IEM for iconic edge model). For example, the edge structure of a B-52 airplane could easily be stored in the same manner that a road network is stored in a GDB. The important distinctions are that (1) the object is not unique but may exist in any number of copies having identical geometry, (2) the object can appear and disappear at many earth locations during certain time lapses, and (3) the location and orientation of the object is not known a priori even after imagery is registered with a GDB.

4.1 Registration of image structures to models

Image registration and object detection are treated below as two sides of the same coin. A technique is developed for matching image structures with iconic structures in a stored map (GDB) or model (IEM). Abstractly there is no difference between a map and a model and the algorithms discussed in 4.2 make no distinction between the two. Since registration and object detection have traditionally been distinct concepts background work is discussed under separate categories.

4.1.1 Image registration

The problem of registering information from one image to that of another image or map, possibly made at a different time and from a different perspective, is one of the major problems in image processing. There are many applications. In medicine there is the problem of comparing two x-rays made a year apart. In industrial engineering there is the problem of inspecting an assembly of parts to check conformity with a blueprint. In photo interpretation there are the problems of collating information about a single point from a set of images from different sensors and of analyzing a given area for changes over time.

Mathematically formulated, the problem of registering two images is the problem of determining a transformation T that maps arbitrary point P_1 in the first image coordinate space to corresponding point P_2 in the second image coordinate space. The work discussed in this report assumes flat imagery and linear transformations; that is, points have 2 coordinates (x,y) and T is specified by a rotation θ and translation (xs,ys). In general points may be specified by more than 2 coordinates and transformation T could have many parameters if non-linear warping is required.

A straightforward registration technique is to use human selection of corresponding "control" points in the two images. These points usually represent salient features such as the corners of buildings or the intersection of roads or streams. Let transformation T be defined by a parameter vector α and let the set of corresponding points be $\{(P_{11},P_{21}), (P_{12},P_{22}),\ldots,(P_{1k},P_{2k})\}$. The best transformation T_α mapping the first image into the second can be defined as that transformation such that

$$\varepsilon=\sum_{i=1,k} d^2(T_\alpha(P_{1i}),P_{2i}) \text{ is minimized}$$

where d^2 is the squared distance between the control point p_{2i} in the second image and the corresponding control point p_{1i} transformed into the second image coordinate space. Thus, once corresponding control points are chosen classical least squares fitting can produce the "best" transformation to be used to register all points.

Automation of control point selection automates the entire registration procedure. Van Wie and Stein [1977] have reported on a system which brings ERTS-LANDSAT imagery into registration with UTM* maps. Map control points are human selected and so are the control points of the first imagery processed. Binary gradient masks M_i in the neighborhood of each image point P_{1i} are computed and stored with the map control points P_{2i}. When subsequent imagery of the area is obtained via the satellite the stored gradient masks M_i are correlated with the gradient image in order to locate the points P_{1i}. This technique depends on the justifiable assumption that the registration transformation for this problem is approximately known so that the correlation can be done in a focused manner, i.e. the orientation and neighborhood for correlating the masks is approximately known. According to Van Wie and Stein this automatic control point selection goes well about 80% of the time and requires human intervention about 20% of the time because of weak correlation.

The work of Horn and Bachman [1977] deals with the registration of an aerial image with a synthetic image constructed from a model of the light source (sun) and ground elevation. The best transformation T_α is gotten by hill-climbing (optimization) in the α-parameter space to maximize a criteria function which considers <u>all</u> points corresponding in the two images and not a selected set of control points. Such a procedure depends on having a good approximation of the optimum α to begin

*Universal Transverse Mercator

the optimization and requires a large computational effort at each of its steps in order to determine fitting quality of T_α. Because it is a global procedure and requires no specially selected features it should be robust in performance.

Some middle ground has been explored at SRI by [Barrow et al 1977]. An image is characterized by a sparse set of feature points which can be automatically extracted. Points along lineal features of good contrast are recommended to represent the image. Similarly, points of lineal features (i.e. a coastline) are represented in a map. Hill-climbing optimization from an assumed approximate α is used to determine the transformation T_α which is optimal for matching points in the image with points of the map. Determining the quality of fit for a given T_α is aided by a technique known as "chamfer matching". Computationally this method is far more palatable than that of Horn, but it does rely on feature detection and does not accurately evaluate the fit of T due to the chamfering trick. Faster but perhaps less reliable results should be expected.

Section 4.2 of this report presents a novel approach to the registration of images using straight edge content. Clusters are formed in an α-parameter space by <u>distributing</u> points gathered from local evidence. The parameter space is the space to be "searched" for the optimal registration transformation T_α. An item of local evidence is a pair of edge elements, one from the first image and one from the second image, which could be interpreted as being the same feature. The correct transformation parameter set α is that set that <u>integrated</u> the most local support.

4.1.2 Object detection

Object detection has been handled in various ways in past work. The most obvious technique is template matching which can be done in the Fourier domain to achieve rotation, translation, and scale invariance. Duda and Hart [1973] and Rosenfeld and Kak [1976] describe 2-D template matching and Zahn and Roskies [1972] describe matching of invariant features gotten from the 1-D Fourier expansion of the boundary of the object. Structural pattern recognition attempts to recognize objects as a synthesis of parts by either ad hoc or formal techniques. The work of Guzman [1971] and Shaw [1970] are examples in this category.

Template matching has as its strength the ability to integrate over many pieces of evidence, usually points, to reach a match and is quite tolerant of imperfect or noisy input. While conceptually simple template matching can be expensive to implement digitally. Synthetic techniques make a low level interpretation of the data in preprocessing and then usually sequentially interpret the low level primitives to assemble objects. While relatively efficient in digital implementations sequential synthetic techniques are difficult to control in situations where data can be imperfect or noisy.

A third object detection technique is conceptually described in [Duda and Hart 1973, Sec 12.3] and practically implemented by Perkins [1977]. This technique involves determining a transformation that would map the image, or part of it, to a model of the object and is a 3-step process.

(G1) Obtain corresponding structures in the image and model. Structures correspond when they have the same shape, size, etc. Image structures are to be obtained automatically while model structures may be interactively compiled.

(G2) Determine transformation parameters $\alpha=(a_1,a_2,...,a_n)$ such that T_α maps at least some corresponding image structures (points, lines, arcs, etc.) onto model structures.

(G3) Determine the degree of match between the entire set of transformed image structures and model structures.

4.2 A new procedure for registration and object detection

This report describes a new technique for performing the 3-step process described above. The technique is a hybrid of template-matching and structural analysis and combines the advantages of those two procedures. The specific interpretation of the general steps above are as follows.

(S1) Assume all structures of the same type correspond. For example, assume each straight line segment in the image can correspond to each straight line segment of the model, each convex curve in the image can correspond to each convex curve of the model, etc. For each pair of structures (s_i,s_m), where s_i and s_m are structures from the image and map respectively, compute transformation parameters α and place a unit of measure in α parameter space.

(S2) Possible transformations T_α between image and model are detected as clusters in α parameter space formed in step S1 because heavy measure in α space means that many correspondences are explained by T_α

(S3) Evaluation of the match strength of each T_α from step S2 is gotten by either computing an average distance between all corresponding structures or by counting the number of image structures explained by the model structures under T_α.

It must be noted that the procedure just outlined is essentially high-level Hough detection -- distribution of mass to a parameter space to detect evidence of global structure.

4.2.1 An illustrative example

A simple example of this process is illustrated in Figure 24. Assume that the image can be represented by the 4 directed edge elements shown in (a) while the map contains the edge elements in (b). It is assumed that the length of the edge elements is accurately known. There are 16 possible ways that an edge element from (a) can be paired with an edge element in (b). Each pairing yields unique transformation parameters (θ,xs,ys) as shown in (c). Four of the 16 possible pairings yield a consistent interpretation -- rotate by θ=0.79 radians and translate by (4.5,2.0). The parameters from the 4 correct pairings form a cluster in α=(θ,xs,ys)-space while the parameters from incorrect pairings are sparsely distributed in the space. In practical cases there will be many more than 4 primitive structures and not all pairings will be possible (i.e. due to size or shape differences) so the presence of a cluster in the parameter space should be even more obvious.

4.2.2 Clustering techniques

Two different clustering techniques were used for L.N.K. registration and object detection experiments. They shall be called the hierarchical technique and the variable resolution technique. In the hierarchical technique clustering was first done on θ alone and then in (xs,ys)-space given a fixed θ hypothesis. This technique was useful for development and human interaction because θ-space could be viewed as a histogram and (xs,ys)-space as a scatter plot. Points of high density in the histogram or scatter plots were found either automatically or by human selection.

In general, true endpoints of edge elements cannot be gotten reliably as was assumed in the example of Figure 24. However, good corner points can be gotten if correct pairs of edge elements are extended to an intersection. These corner points can then be used to accurately define the registration transformation. Details are as follows. For each pair (s_i, s_m) of image edge elements and map edge elements, the rotation θ_{im} necessary to rotate s_i into s_m is recorded in a histogram. (A clustering space of 360 one-degree bins.) The top 3 peaks of the histogram, after smoothing, are examined further. For a given θ, points in (xs,ys)-space are gotten as follows. Let map edge elements $C_m D_m$ and $C_n D_n$ specify a ground control point (GCP) in the map. For each image edge element pair $A_i B_i$ and $A_j B_j$ rotating into $C_m D_m$ and $C_n D_n$ by rotation θ, a translational component (xs,ys) is readily computed and a unit of measure placed in (xs,ys)-space. When all GCP's are treated in this manner, clustering in (xs,ys)-space is easily done by examining a scatter plot and a full set of transformation parameters $\alpha=(\theta,xs,ys)$ is available. The goodness of T_α can then be evaluated as in step S3 above.

The variable resolution clustering technique uses a fixed 10x10x10 partition (binning) of the 3-D parameter space. After smoothing, heavy regions of 2x2x2 bins are selected for clustering at the next level. Points in the 2x2x2 bin region are redistributed to the 10x10x10 bins by scaling and cluster detection reapplied. This refinement process is continued until there is no more cluster evident or until the bin size reaches the limit of resolution inherent in the problem domain. Since the size of a bin is reduced to 1/5th size with each level of clustering, 3 levels is typical yielding 3° x 2 pixel x 2 pixel bin size for a (θ,xs,ys) originally constrained to a 360° x 500 pixel x 500 pixel space. Note that only rotation and translation parameters are used: transformations using scale changes as well would involve 4 parameters instead of 3 and were saved for future experiments.

4.3 Image registration experiments

Several image registration experiments were run in order to test the viability of the new concept. The hierarchical clustering technique described earlier was used and there was considerable human interaction in the early stages of testing. Input to the procedure were two sets of directed straight edge elements; one called the image and the other called the map. The image edge set was extracted automatically by the Hough detector described in Section 2. The map edge set was compiled by humans making measurements on print plots of the digital imagery. The resulting maps tended to be very crude because 1) the amount of human effort in making measurements was great and 2) a procedure was desired that would perform well on crude input which would be expected from automatic processing. Several edge pairs whose intersections made good ground control points (GCP's) were identified in each map. Typically GCP's were chosen as nearly 90° intersections of large edges. GCP's were used in the second stage of hierarchical clustering to determine the translation (xs,ys). Output from the registration procedure was a set of possible transformations (θ,xs,ys) mapping the image edge set onto the map edge set and strength value for each. The strength of a registration transformation $T_\alpha=T_{(\theta,xs,ys)}$ was determined heuristically and left for human interpretation. Three numbers were actually output. For each pair of edge elements (s_i,s_m), s_i from the image and s_m from the map, a value in the range [0,1] could be assigned according to how close $T_\alpha(s_i)$ was to s_m in the map coordinate space. The procedure reported a count of all image edges s_i and a count of all map edges s_m which had no corresponding edge element with above 0 value. The sum of all match values was also reported.

4.3.1 Image registration data sets

Three major test sets were used to test the registration procedure. The images used were AFB2,GAFB, and UNB (shown in Figures 11 - 13). Edge element sets automatically extracted from these images are shown in Figures 11, 14, and 16. The maps for AFB2 and GAFB were made from print plots of the digital imagery making it known a priori that the correct registration transformation was (θ=90°,xs=0,ys=0). Since repeat coverage was available for UNB, image edge elements were extracted from UNB1 (Figure 13) and a map was constructed from print plots of UNB2 (Figure 7). Three ground control points were used in each UNB image to establish what a good approximate registration transformation should be. Plots of the edge elements used for AFB2,GAFB and UNB are shown in Figures 25,26 and 27 respectively. Details of the 3 test sets are summarized in Table 1.

4.3.2 Results of the image registration experiments

Table 2 summarizes the results of the experiments with the test images described in Table 1. The results confirm the usefulness of the new registration concept and there should be additional comfort in the fact that very crude maps were used and other useful registration heuristics were ignored. It is important to point out that clustering in θ-space was done automatically while clustering in (xs,ys) was done interactively. The histogram of θ rotations formed by rotating each image edge into each map edge was first smoothed by a 3° averaging window and then subjected to peak detection. Up to 3 of the best peaks were passed on to the next stage of clustering. The threshold used was 50% of the number of edge elements from the image. Clustering in (xs,ys) space was done interactively by human examination of scatter plots presented by computer. Evaluation of the strength of selected transformations (θ,xs,ys) were computed automatically.

4.3.2.1 AFB2 data set

As Table 2 shows, there were two strong responses in the interpretation space (θ,xs,ys) for AFB2 registration. Under the "correct" interpretation (θ=90°,xs=2, ys=2) 35 of 48 image edge elements aligned with 13 of 16 map edge elements for a total strength of 25.67. Due to the symmetry in the building patterns of AFB2 an incorrect interpretation (θ=270°,xs=-46,ys=-18) aligned 15 of 48 image edge elements with 4 of the 16 from the map for a total strength of 12.71. This type of ambiguity was observed to some extent in all of our imagery because man-made structures exhibit perpendicularity which results in 90° peak displacements in the θ-space of our procedure.

4.3.2.2 GAFB data set

Figure 28 a shows a plot of smoothed strength in θ-space. There are 8 strong peaks, one for each map edge element, but only the peaks at 90° and 315° draw support from at least half of the image edge elements. A peak is observed for each map edge element because of heavy image activity in the direction of 277°. There are 45 image edge elements with direction 277° ± 3°. Under a 90° rotation these will align with map edge #8 which has direction 4°. The correct 90° interpretation is also supported by 28 other image elements aligning with one of the other 7 map edge elements for a total support of 73 in θ-space. However, the alignment of the 45 image elements of direction 277° with map element #6 (direction 230°) supports the interpretation θ=315° with strength 45. Random support from other alignments bring the support for θ=315° to 63, just above the threshold. While the 2nd stage of clustering removes the contention from θ=315° it is desirable to reduce ambiguity in the θ-space. This can be done by construction of a more comprehensive map for GAFB so that most of the image edge elements would support the correct rotational interpetation and hence enhance its strength relative to alternatives. Figure 28b shows clustering in (xs,ys)-space under the hypothesis that θ=90°. Figure 28c shows a scale refinement of part of Figure 28b which enabled a choice of cluster center (xs=2,ys=3) and an evaluation of support for (θ=90°,xs=2,ys=3) to be 30.15. No cluster for θ=315° exhibited any strength relative to the (θ=90°,xs=2,ys=3) transformation.

4.3.2.3 UNB data set

Table 2 shows that our procedure uncovered a very good approximate registration transformation as the strongest interpretation, exceeding all others by at least a factor of 2. That transformation, (θ=330°,xs=-142,ys=8), aligned 19 of 200 image edge elements with 12 of 22 from the map. The strongest contender, (θ=237°, xs=453,ys=-433) aligned only 6 image edge elements with 2 from the map. We believe that construction of a more comprehensive map would have enhanced these results just as it would have in the GAFB case.

4.3.3 Discussion of the image registration experiments

The results indicate that the registration concept introduced in Section 4.2 can be used to register image edges to model edges. Because global interpretations are formed by integration, supporting local evidence can be incomplete or in error and the correct interpretation can still be obtained. Moreover, the process of accumulating support is, unlike many synthetic techniques, independent of the order in which local evidence is considered. In the experiments the correct interpretation always dominated incorrect interpretations in strength of support.

Unlike other registration techniques the new technique need make no assumption that an approximation to the transformation parameters be known a priori. Thus this procedure is a candidate front-end procedure to more sophisticated non-linear procedures which require a good start in their hill-climbing. More general transformations can be handled in a similar manner. However, it must be pointed out that clustering techniques, such as the Hough transform, lose efficiency when the size of the parameter space increases. The simplicity of the new technique makes it quite attractive for rotations and translations. The complete GAFB image was represented by 124 straight edge elements, or 124x4 integers, while the GAFB map was specified by only 8 edge elements.

The results reported in Table 2 were gotten with automatic peak detection in the θ-space and interactive cluster detection in the (xs,ys)-space. The (xs,ys) -space clustering was then completely automated and similar results were obtained. The ambiguities arising in images with high culture content make the hierarchical clustering procedure suspect. The ambiguity present in Figure 28 a is rather alarming. Combination of edge elements (i.e. to get Figure 15 from Figure 14 improved this situation considerably. However, further experiments in object detection showed that many of the highest peaks in θ-space could be false and that 3-D clustering is really necessary. The next section discusses the more robust clustering technique in the framework of object detection.

4.4 Object detection experiments

Section 4.2 introduced a new concept for registration which used clustering to form a globally valid registration transformation from local evidence of matching structures. Experiments in registering images to maps were discussed in Section 4.3. Although object detection is the primary concern of the present section it should be remembered that techniques applicable here will also apply to general registration problems.

Three types of structures have been used in L.N.K. research as primitive structures for registration. They are all derivable from the output of edge extraction. Extraction of primitives was discussed in Section 2. Since the registration procedure is quite tolerant of errors the feature extraction procedures themselves need not be highly refined. The three structures used for registration are as follows.

(E1) Straight edge elements

Straight edge elements indicate a straight side of an object and are represented by endpoints A' and B' as in Figure 29 Traversal of the edge from A' to B' keeps the darker halfplane to the right. Model edges are constructed by human or interactively and are assumed to have accurate endpoints. Image edges are gotten automatically and endpoints are assumed to be inaccurate.

(E2) Points of sharp curvature

Points of high curvature on curved edges are typed as convex or concave according to whether the inside of the curve is darker or lighter than the outside. The type is easily coded as a sign on the curvature as the curve is traversed with the darker region to the right.

(E3) Points of angular intersection

Two straight edges intersecting at a point forming a blunt angle create an intersection point which may be typed according to the angular size, i.e. 60°-120°,90°,etc. Intersection points, are easily gotten by considering nearby pairs of detected line segments as in E1.

4.4.1 Determining transformation T_α from only straight edge correspondences

Let points A and B define an image edge structure and let points C and D define a map edge structure. Let (θ_i,r_i) and (θ_m,r_m) respectively be the polar form of the halfplanes determined by edge AB in the image coordinate system and edge CD in the model coordinate system. We compute α, or a set of α, so that edge AB maps onto edge CD under transformation T_α. Only rotation and translation will be allowed here so that $\alpha=(\theta,xs,ys)$. Given (θ_i,r_i) and (θ_m,r_m) θ is easily determined as $\theta_m-\theta_i$. Once edge AB is rotated by $\theta=\theta_m-\theta_i$ to get edge A'B' the configuration of Figure 29 is obtained. The translational part of α is then constrained by the equation.

$$\Delta x \cos \theta_m + \Delta y \sin \Delta_m + (r_i-r_m)=0.$$

Actually, since A'B' is a finite line segment $\Delta x,\Delta y$ are further constrained because point A' should translate no further than point C and point B' should translate no further than point D. (Some tolerance can be given since the edge detector can overshoot corners.)

The result is that the correspondence of image edge AB and model edge CD yields a <u>line segment in α - space</u> between α points (θ,xs_1,ys_1) and (θ,xs_2,ys_2). If all edges are paired with all model edges, α - space will be sparsely filled with line segments from incorrect correspondences but will contain a cluster of intersecting line segments for the set of correct edge correspondences.

Figure 30 shows a set of model edges defining an airplane. These were gotten by human measurement of a blowup of a grey-scale picture of an airfield. Figure 31 shows a set of image edge elements extracted from a 512x512 area of the airfield.

Figure 32 is a plot of the line segments in that slice of α - space with $320 \leq \theta \leq 340$. The rest of α - space is quite barren. From the data shown in Figure 32 transformation parameters α=(θ=330°,xs=-104,ys=42) were gotten by a simple binning implementation of clustering. Under T_α only one edge of the 10 model edges (the back of the tail) was not matched to image edge data. Further experiments of this kind are detailed in Section 4.4.3.

4.4.2 Determining transformation T_α from correspondences of abstract edges

As Figure 32 shows, α - space can be quite noisy due to incorrect correspondences and to too little constraint on xs,ys. Both noise sources can be significantly reduced if the lengths of corresponding edge structures are forced to agree. Thus the endpoints of the edge structures must be reliably delimited. To achieve this, point structures can be detected and abstract edges formed by spanning pairs of points. These edges, assumed now to have accurate length, can be used for registration exactly as discussed in Section 4.2.

A specific case is treated here but the general concept should be obvious from this example. Abstract straight edges can be formed by using an intersection point as the vector tail and a high curvature point as the vector head. Figure 33 shows the model of the airplane under this new scheme. There are 6 intersection points (4-wing-fuselage intersections and 2 tail-fuselage intersections) and 5 points of high curvature (2 wing tips, 2 tail tips, and the nose) yielding 30 abstract edges. Compare Figures 33 and 30.

The straight edge content of a second 512x512 area of the airfield image is shown in Figure 34. Points of intersection were gotten from this data and combined with high curvature points extracted by another algorithm to get the set of abstract image edges plotted in Figure 35. Note the confusion created by the presence of incomplete structures from 2 airplanes. The clustering succeeded, nevertheless, in detecting a transformation T_α that mapped 31 abstract image edges onto 16 abstract model edges. Figure 36 gives a plot of only those edge elements from Figure 35 that matched edge elements of the model. Note that the 31 abstract edge elements represent only 16 true edges. There is a duplication of abstract edges because several edge detections along straight edges caused duplication of intersection points as is evident in Figure 35. Merging copies of the same intersection would have been an improvement but was felt to be unnecessary.

4.4.3 Experiments in object detection

The procedures illustrated in Sections 4.4.1 and 4.4.2 were tested on several windows of the AFB image (Figure 11). Since the airplanes were known to be of rough size 256x256 pixels, testing a set of overlapping 512x512 pixel windows was guaranteed to consider all relevant areas. 7 windows were required to cover the entire plane parking area: two of these windows in fact contained no planes and two windows contained all or part of two planes. The nose and wing tips of each plane were identified by a human and a "correct" reference transformation was determined. Actually the transformation was computed in 3 ways so that a measurement error could be established. Table 3 contains a summary of the detection experiments. The centers of the windows are listed in Column 1 while the human computed registration transformation, relative to the window centers, is given in Column 2. For example if all points of the window centered at (256,1536) were shifted by x'=x-256,y'=y-1536 then a transformation of (θ=330°,xs=-55,ys=69) would align airplane #4 with the model in Figure 30.

Results of automatic object detection via registration are given in Columns 3 and 4 of Table 3. The parameters of possible registration transformations are given at the left of Columns 3 and 4 and the quality of match achieved is evaluated at the right of Columns 3 and 4. For example, the best possible detection of airplane #4 in sub-window 6 using straight edge evidence occurs with transformation (θ=332°,xs=-59,ys=66). Under this transformation 18 out of 70 edge elements from the image map onto 9 out of 10 edge elements in the model with an average match value of 63.30 (out of a possible 100). This heuristic average match value is in fact better than that achieved with the human determined transformation which is evaluated in the starred row of Columns 3 and 4. Using abstract edge evidence the strongest cluster in α - space was detected at (θ=332°,xs=-61,ys=61). This transformation aligned 52 of 155 abstract image edges with 10 of 30 abstract model edges. The average of the 52 match values was 29.06; not great but a bit higher than the average match for the human computed transformation.

A binary decision PLANE versus NO PLANE was not made. This is consistent with our philosophy that other information should be integrated into the final decision. Search for unmatched model edges to verify a possible detection is discussed in Section 5 of this report. It can be seen from Table 3, however, that no detection will be falsely dismissed with the current stage of processing. There are some false alarms appearing at this stage -- their removal will be treated in Section 5.

It is interesting to note the ambiguities revealed in Table 3. For instance, in subwindow 2 a fairly strong detection was made with θ=145°. This is 180° off of the correct rotation and shows that the airplane does have some rotational symmetry. While fewer edge elements align with the 145° rotation the average match value of the aligning edges is higher than for the correct transformation. Registration of subwindows 3,4 and 5 shows that some 90° and 270° rotational symmetry also exists. The most valuable information for removing the ambiguity lies in the edges of the tail which apparently are the most difficult to extract bottom-up. Top-down testing for previously undetected tail edges is discussed in Section 5.

4.5 Discussion of registration results

We interpret the results to be highly promising and to have demonstrated the features of the new registration procedure.

. The procedure is fairly efficient since it uses only cheaply extracted primitive features of imagery. Recall that only 124x4 integers were used to represent the GAFB image and only 8x4 integers represented the GAFB map.

. The procedure can operate with fair amounts of missing information or irrelevant background. See Figures 30 to 32 for example.

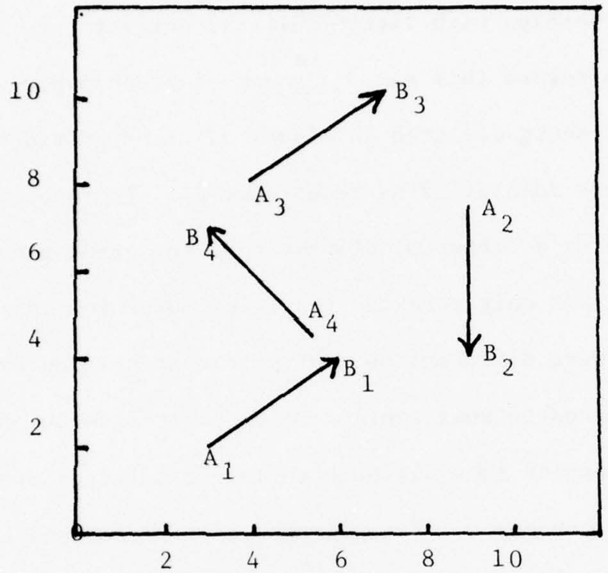
. The procedure does not require a starting approximation to the transformation sought. In our experiments we have assumed only that the rotation θ was confined to [0°,360°) and that xs and ys were within half the window size in either direction. In particular such freedom enables the procedure to be used for detection of moveable objects--objects in aerial imagery or parts on an assembly line.

The cost of the new procedure should be considered in more detail. The chief cost component is the extraction of the image structures, i.e. the edge elements or points. Fortunately, hardware can be used for extraction of straight edges [Stockman 1977]. If there are i edge elements from the image and m from the map then i m pairs are formed for all possible structure correspondences. If the edge element length is reliable almost all of the i m pairs are rejected immediately while the remaining few are used to define points in α - space. Clustering is done on f i m line segments where $0 \leq f \leq 1$. The clustering procedure uses 10x10x10 bins so the cost of smoothing and peak detection is constant c_1 while the cost of filling the bins varies linearly with the number of possible edge correspondences $c_2 fim$. Clustering is typically repeated 3 times until the bin scale is comparable to the accuracy obtainable. Thus the total cost is $3(c_1+c_2\ fim)$. Since m can be controlled to be a small fixed number the total cost can practically be conceived as linear in i, the number of image edge elements. Since not all of the edge elements need be used to establish registraticn some scattered sampling of them could be used and thus i could be controlled as well.

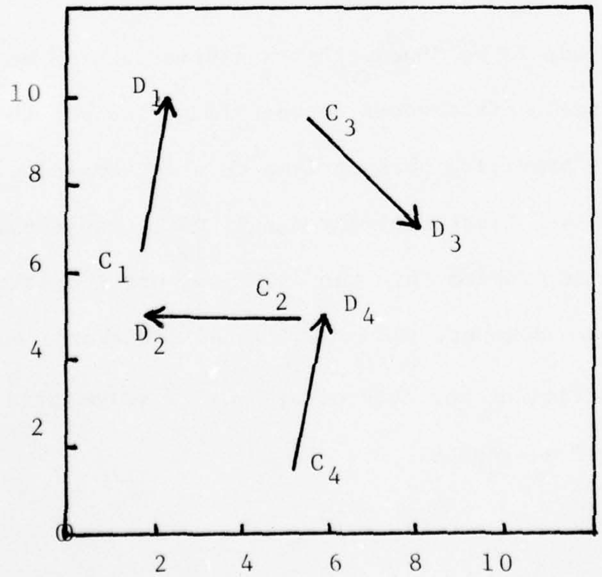
Registration of large areas is aided by the presence of map features over wide regions of the map. Object detection on the other hand requires separate consideration of locally confined sets of edges. If the object fits in a square of side s we examine all 2sx2s areas with endlap and sidelap of s. This guarantees that every object will be enclosed in at least one area. As Figure 34 shows more than one object can be in one area. Because many registrations have to be done to detect small objects, global registration of the general area with a base map is worthwhile in order to delimit what specific areas are to be examined for which specific objects. Planes are sought on airstrips, and ships are sought in water, for instance.

More work needs to be done in testing the use of abstract edges in registration of varied terrain, perhaps with little cultural activity, to archived maps. A pilot experiment was done toward this end. A sheet of graph paper was overlaid on the feature map of Harrisburg depicted in Figure 17 and prominent intersection and curvature points were identified by researcher #1. The process was then repeated by researcher #2 using a different placement of the graph paper, i.e. a different coordinate system. Not only were the resulting point features in different coordinate systems, but there were different sets of points identified by the two researchers. This situation compares to what would likely result from automatic detection in repeated coverage. Despite some differences in the selected structures there was enough in common to produce a large cluster in parameter space to identify the correct transformation between coordinate systems. Hopefully, this effect can be repeated in future experiments matching aerial imagery to maps.

Further work needs to be done with transformations of more than 3 parameters. For aerial imagery scale differences between image and map should certainly be handled. For images providing perspectives on a 3-D world up to 6 parameters might be required. Use of 4-D clustering for a full RS&T transformation does not seem to present a much greater problem than the 3-D case, particularly if a hierarchical approach can be used. However, the simple local structural correspondences currently being used are insufficient for determination of 6 parameters with subsequence placement of mass in a 6-D α - space.


10
8
6
4
2
0
2
4
6
8
10
B_3
A_3
A_2
B_4
A_4
B_2
B_1
A_1

(a)


10
8
6
4
2
0
2
4
6
8
10
D_1
C_3
D_3
C_1
C_2
D_4
D_2
C_4

(b)

i	j	A	B	C	D		θ	xs	ys	
1	1	3.0,2.0	6.0,4.0	1.7,6.4	2.3,10.0		0.82	1.1	2.8	
1	2	3.0.2.0	6.0,4.0	5.3,5.0	1.8,5.0		2.55	8.9	5.0	
1	3	3.0,2.0	6.0,4.0	5.5,9.5	8.0,7.0		4.91	3.0	12.1	
1	4	3.0,2.0	6.0,4.0	5.1,1.5	5.8,5.0		0.79	4.4	-2.0	*
2	1	9.0,7.5	9.0,4.0	1.7,6.4	2.3,10.0		2.98	11.8	12.3	
2	2	9.0,7.5	9.0,4.0	5.3,5.0	1.8,5.0		4.71	-2.2	14.0	
2	3	9.0,7.5	9.0,4.0	5.5,9.5	8.0,7.0		0.79	4.4	-2.2	*
2	4	9.0,7.5	9.0,4.0	5.1,1.5	5.8,5.0		2.94	15.4	7.1	
3	1	4.0,8.0	7.0,10.0	1.7,6.4	2.3,10.0		0.82	4.8	-2.0	*
3	2	4.0,8.0	7.0,10.0	5.3,5.0	1.8,5.0		2.55	13.1	9.4	
3	3	4.0,8.0	7.0,10.0	5.5,9.5	8.0,7.0		4.91	-3.1	11.9	
3	4	4.0,8.0	7.0,10.0	5.1,1.5	5.8,5.0		0.79	7.9	-7.0	
4	1	5.5,4.5	3.0,7.0	1.7,6.4	2.3,10.0		5.33	-5.2	8.3	
4	2	5.5,4.5	3.0,7.0	5.3,5.0	1.8,5.0		0.79	4.6	-2.1	*
4	3	5.5,4.5	3.0,7.0	5.5,9.5	8.0,7.0		3.14	11.0	14.0	
4	4	5.5,4.5	3.0,7.0	5.1,1.5	5.8,5.0		5.30	-1.7	3.6	

(c)

Figure 24. Example of global registration via clustering of local evidence. Image edge elements in (a) need to be rotated 45° and then translated (4.5,-2.0) to be transformed into corresponding map edge elements in (b). (c) 16 units of measure are amassed in (θ,xs,ys) -space forming a cluster at (θ=0.79,xs=4.5,ys=-2.0).

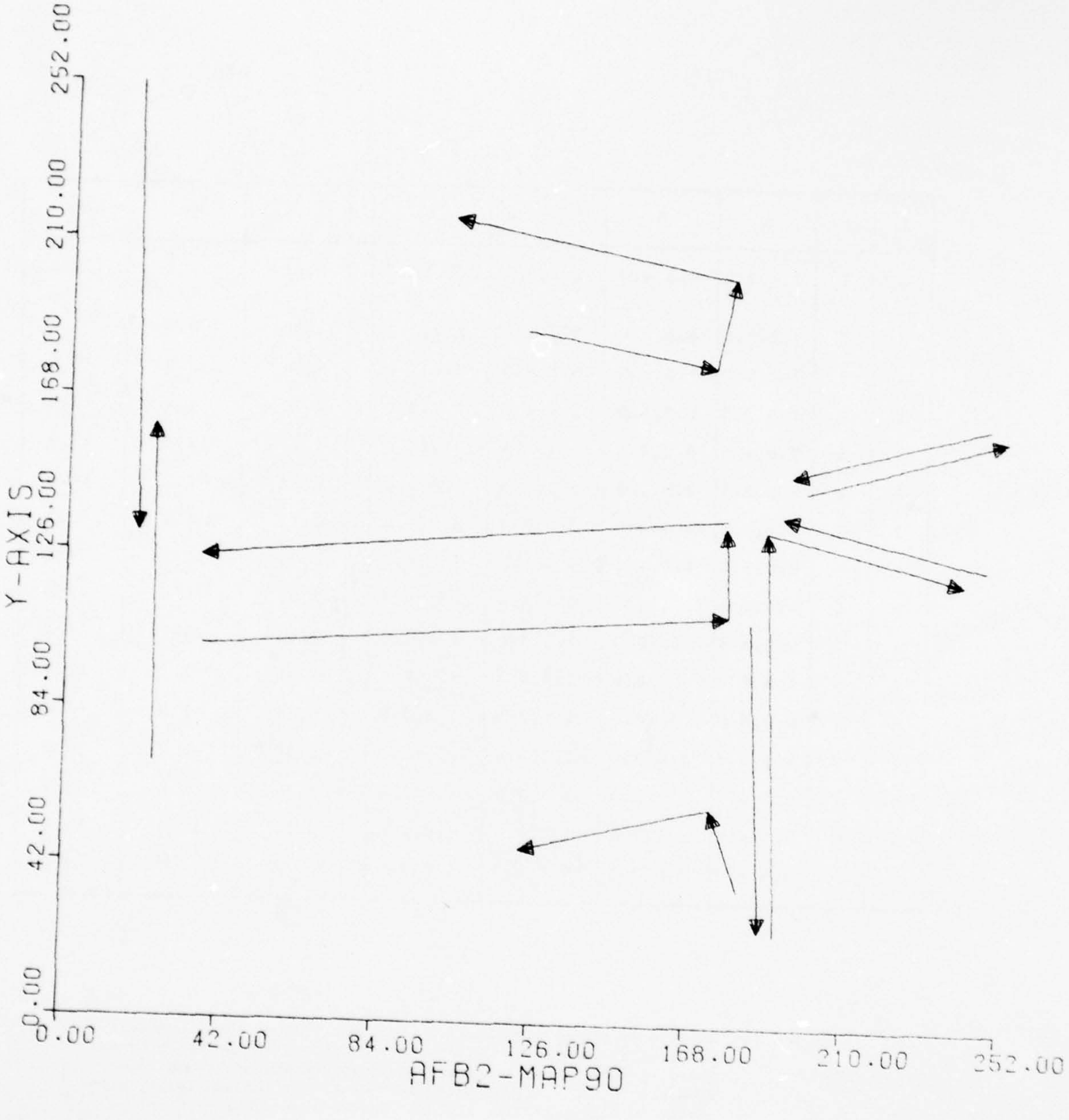


Figure 25. AFB2 map made by human on print plot of digital image.

AD-A067 166   L N K CORP SILVER SPRING MD   F/G 9/4
THE USE OF MODELS IN IMAGE ANALYSIS.(U)
JAN 79 G C STOCKMAN, S H KOPSTEIN   F33615-76-C-0521
UNCLASSIFIED   AMRL-TR-78-117   NL

2 OF 2
AD
A067166

END
DATE
FILMED
6-79
DDC

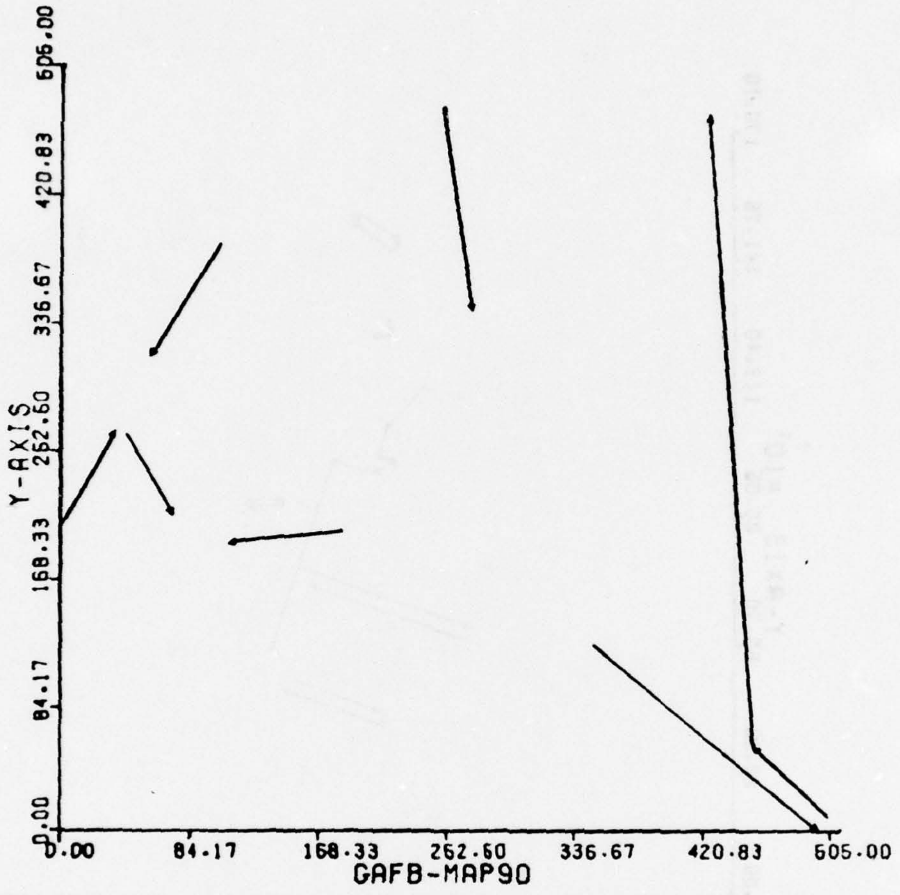


Figure 26. GAFB edge element map made by human on print plotted picture.

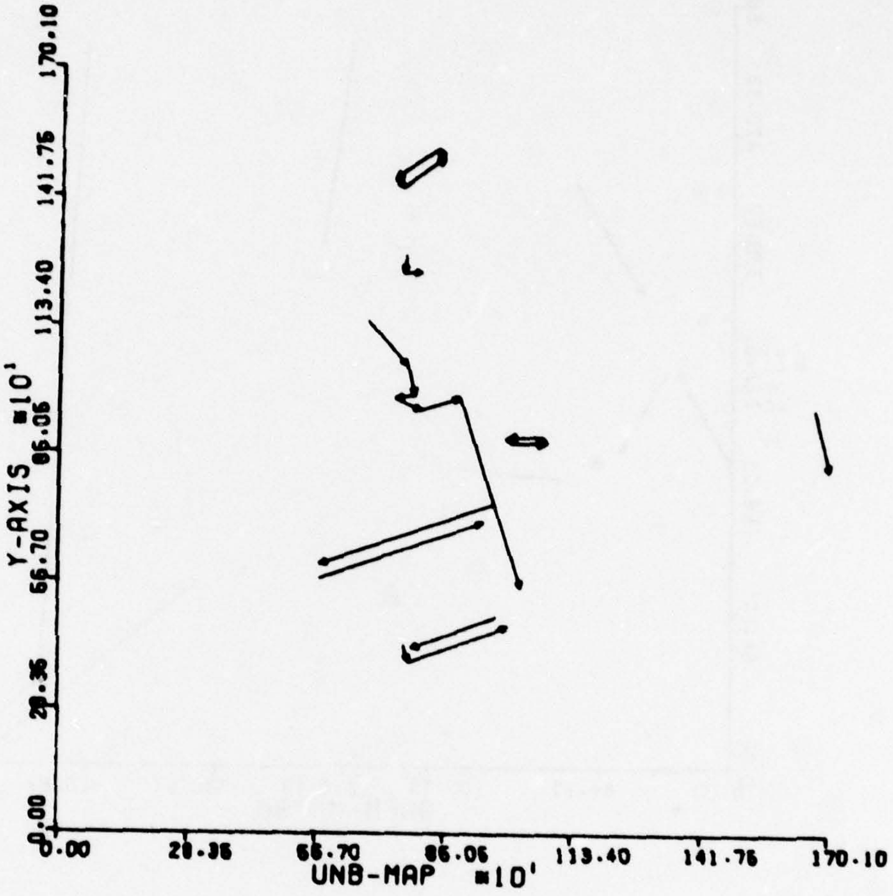
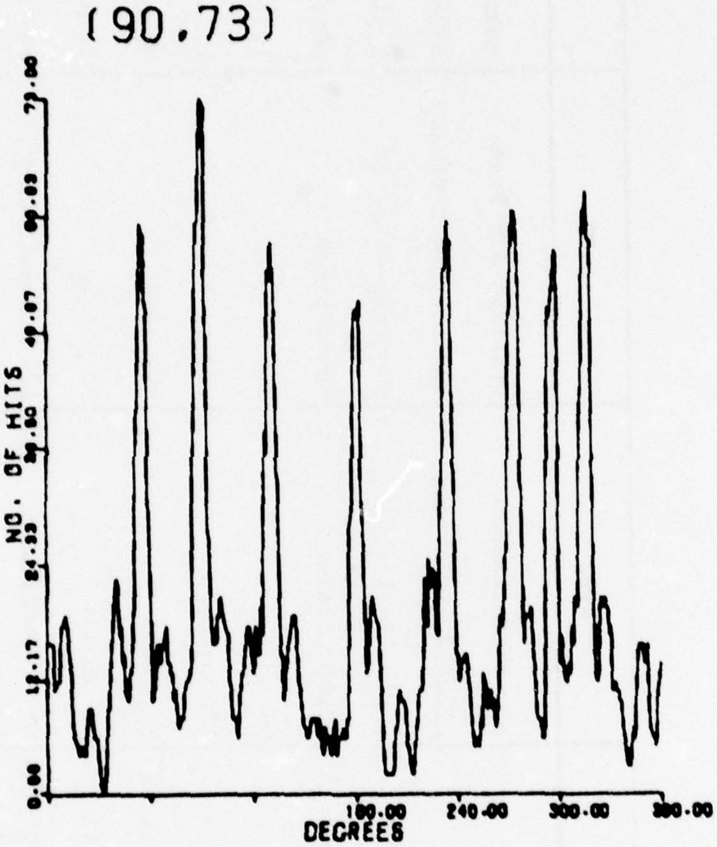


Figure 27. UNB edge element map made by human on print plots of UNB2 image .

TABLE 1

DETAILS OF IMAGE SIZE AND HOUGH DETECTION OF EDGE ELEMENTS FOR THREE EXPERIMENTAL DATA SETS

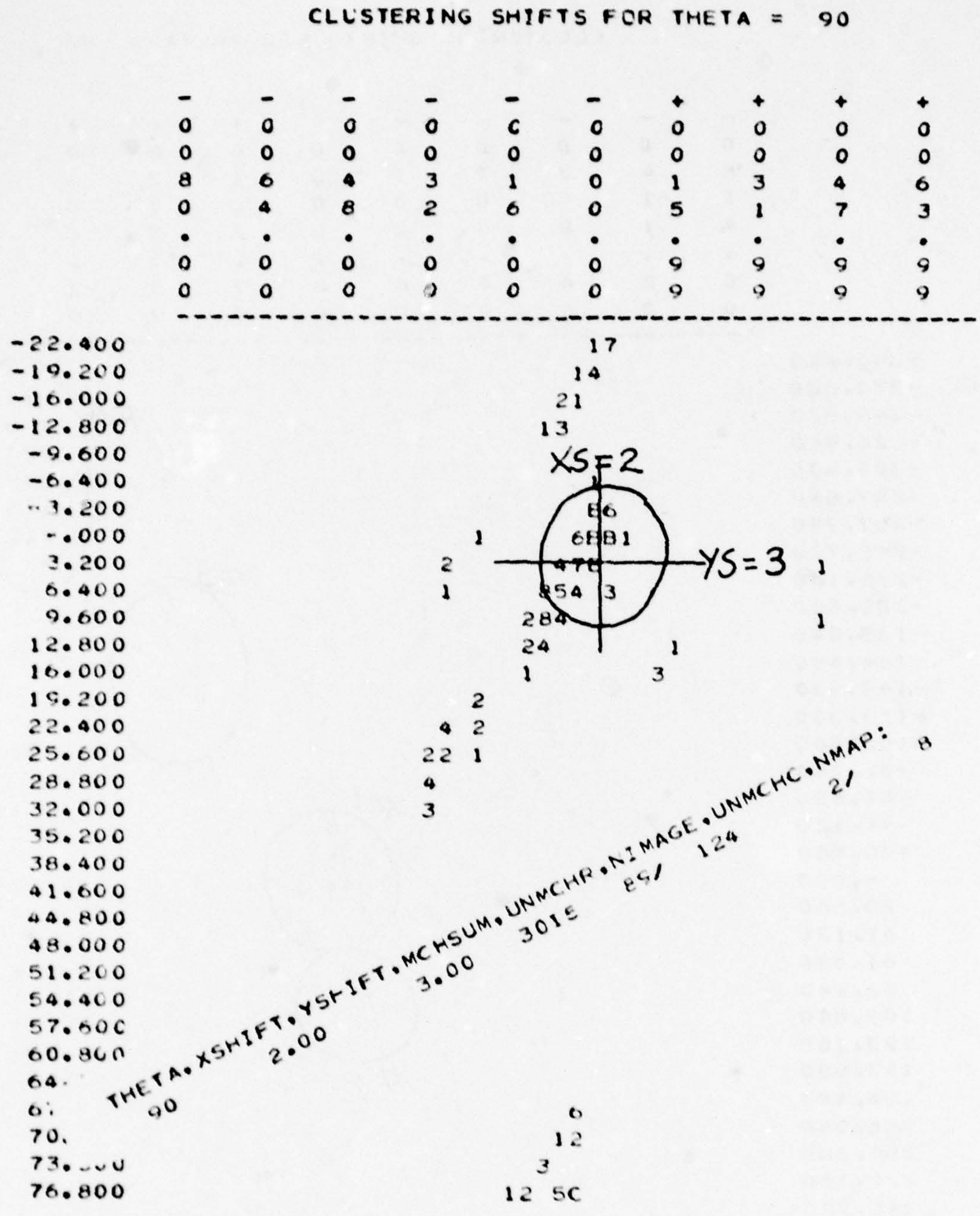
	AFB2	GAFB	UNB
Picture Name	air base closeup	Griffiss Air Base	unknown naval base
Picture size	252x252 6-bit pixels	504x504 6-bit pixels	2208x2208 6-bit pixels
Approximate resolution	1 pixel = 1 sq ft	1 pixel = 4 sq ft	1 pixel = 4 sq ft.
Hough detector size	60x60 pixels	60x60 pixels	60x60 pixels
Number of windows	64	256	1257
Number of pixels along an edge	120 to $120\sqrt{2}$	120 to $120\sqrt{2}$	120 to $120\sqrt{2}$
Total no. pixels used as evidence	5%x60x60	5%x60x60	5%x60x60
Strength threshold for detection	30 pixels	30 pixels	30 pixels
No. image edge elements detected	48 automatically	124 automatically	200 automatically
No. map edge elements constructed	16 by human	8 by human	22 interactively



(a) GAFB: plot of smoothed θ-space.

```
                    CLUSTERING SHIFTS FOR THETA =  90

            -    -    -    -    -    -    +    +    +    +
            0    0    0    0    C    0    0    0    0    0
            5    4    3    2    1    0    1    2    3    4
            1    1    0    0    0    0    0    0    0    1
            4    1    8    5    2    0    2    5    8    1
            .    .    .    .    .    .    .    .    .    .
            0    2    4    6    8    0    7    5    3    1
            0    0    0    0    0    0    9    9    9    9
            ------------------------------------------------
 -390.640                                               3
 -370.080                                              14
 -349.520                                    #4   1 8
 -328.960                                         223
 -308.400                                         232
 -287.840                                         1 5
 -267.280                                         211
 -246.720                                         4
 -226.160                             #3   8
 -205.600                      1               F
 -185.040                      3             9 P
 -164.480                             9 3    I
 -143.920                                    R 3
 -123.360                                      G
 -102.800
  -82.240                                   6L
  -61.680                        #1
  -41.120                         7
  -20.560                        8N
    -.000                       5** 2     2
   20.560                       I
   41.120
   61.680                        *
   82.240            1          6N
  102.800                       6
  123.360
  143.920                        #2
  164.480
  185.040
  205.600
  226.160
  246.720
  267.280                 1
  287.840                 1
```

(b) GAFB: High level clustering in (xs,ys) -space under the hypothesis that θ=90°.



```
                    CLUSTERING SHIFTS FOR THETA =   90

           -    -    -    -    -    -    +    +    +    +
           0    0    0    0    C    0    0    0    0    0
           0    0    0    0    0    0    0    0    0    0
           8    6    4    3    1    0    1    3    4    6
           0    4    8    2    6    0    5    1    7    3
           .    .    .    .    .    .    .    .    .    .
           0    0    0    0    0    0    9    9    9    9
           0    0    0    0    0    0    9    9    9    9
          --------------------------------------------------
-22.400                            17
-19.200                           14
-16.000                          21
-12.800                         13
 -9.600                          XS=2
 -6.400
 -3.200                            B6
  -.000                      1    6BB1
  3.200                   2     476       YS=3   1
  6.400                   1     854 3
  9.600                        284               1
 12.800                        24      1
 16.000                        1      3
 19.200                     2
 22.400                   4 2
 25.600                  22 1
 28.800                  4
 32.000                  3
 35.200
 38.400
 41.600
 44.800
 48.000
 51.200
 54.400
 57.600
 60.800
 64.
 6.                                 6
 70.                             12
 73.                            3
 76.800                       12 5C

THETA.XSHIFT.YSHIFT.MCHSUM.UNMCHR.NIMAGE.UNMCHC.NMAP:
 90      2.00     3.00     3015    89/   124    2/    8
```

(c) GAFB: fine scale clustering.

Figure 28. GAFB registration steps.
(a) Plot of smooth θ-space with best peak at θ=90°.
(b) Coarse scale interactive clustering in (xs,ys) scatter plot.
(c) Fine scale examination of cluster #1 from b) and evaluation of transform (θ=90°,xs=2,ys=3).

TABLE 2

SUMMARY OF REGISTRATION RESULTS
ON THREE EXPERIMENTAL DATA SETS

data set a priori transformation method of determination	AFB2 (90°,0,0) coordinate system change	GAFB (90°,0,0) coordinate system change	UNB (328°,-141,9) matching ground control
	cluster center/radius/strength	cluster center/radius/strength	cluster center/radius/strength
cluster 1	90°/3°/65	90°/3°/73	330°/3°/176
(xs,ys) cluster 11	(2,2)/10/25.67	(223,-183)/5/2.89	(-71,-215)/10/2.35
(xs,ys) cluster 12	(-164,-8)/5/1.05	(2,3)/10/30.15	(-142,8)/10/12.85
(xs,ys) cluster 13	*	(170,-143)/10/6.12	(195,135)/10/5.04
cluster 2	270°/3°/59	315°/3°/63	237°/3°/152
(xs,ys) cluster 21	(-46,-18/5/12.71	(-150,-92/10/0.24	(453,-433)/20/5.14
(xs,ys) cluster 22	*	(-140,-132)/10/0.0	(345,-78)/10/3.48
(xs,ys) cluster 23	*	*	(276,146)/10/3.86
cluster 3	*	*	61°/3°/139
(xs,ys) cluster 31	*	*	(-22,-406)/20/1.28
(xs,ys) cluster 32	*	*	*
(xs,ys) cluster 33	*	*	*

*indicates no viable alternative cluster

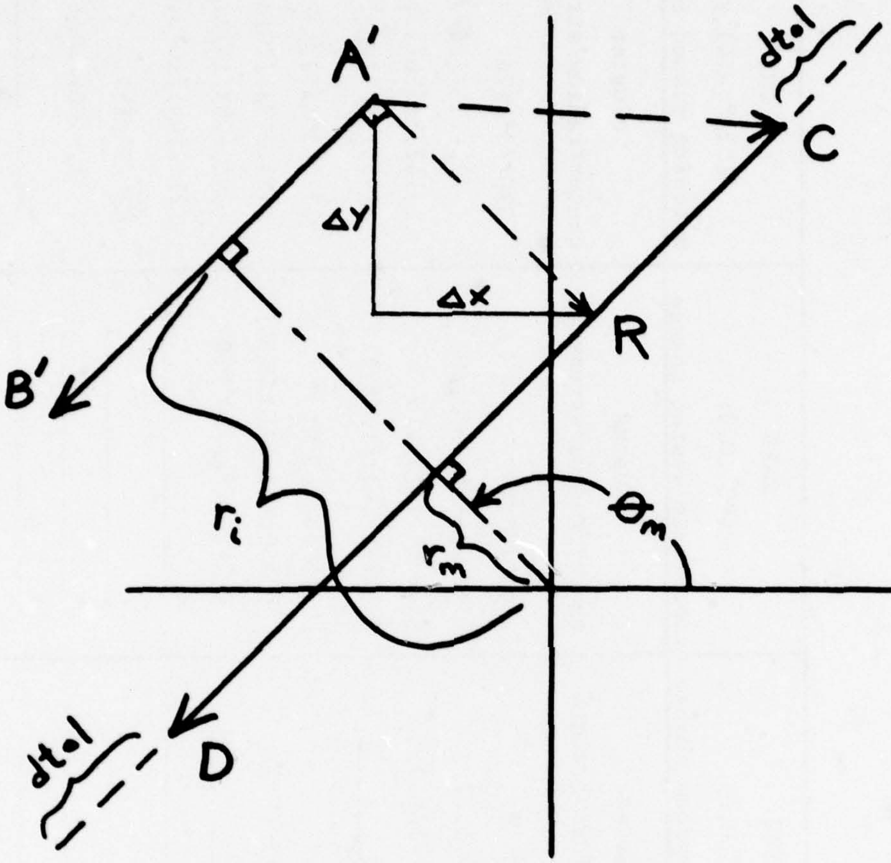


image edge $\vec{AB}$ or (θ_i, r_i) is rotated $\theta_m - \theta_i$ into A'B' or (θ_m, r_i) to be parallel to map edge $\vec{CD}$ or (θ_m, r_m)

unit vector from A' to R is $(-\cos\theta_m, -\sin\theta_m)$. projection of $\vec{A'C}$ onto $\vec{A'R}$ has constant length yielding equation relating Δx and Δy of translation

$$\vec{A'C} \circ (-\cos\theta_m, -\sin\theta_m) = r_i - r_m$$

$$(\Delta x, \Delta y) \circ (-\cos\theta_m, -\sin\theta_m) = r_i - r_m$$

$$\Delta x \cos\theta_m + \Delta y \sin\theta_m + (r_i - r_m) = o$$

Figure 29. Derivation of $\alpha = (\theta, \Delta x, \Delta y)$ mapping image edge element $(\theta_i r_i)$ onto model edge element (θ_m, r_m).

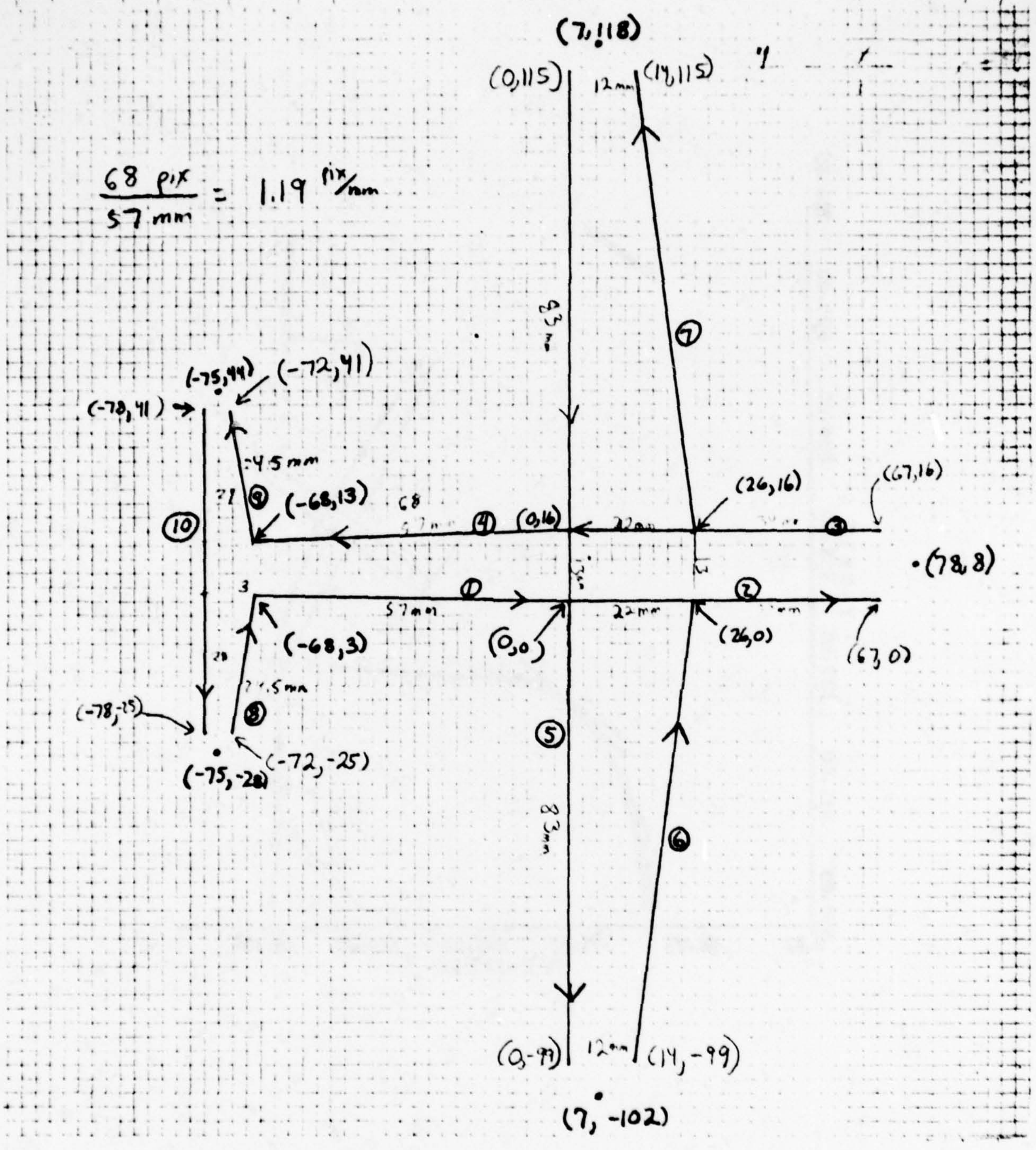


Figure 30. Model of airplane in terms of directed straight edge segments.

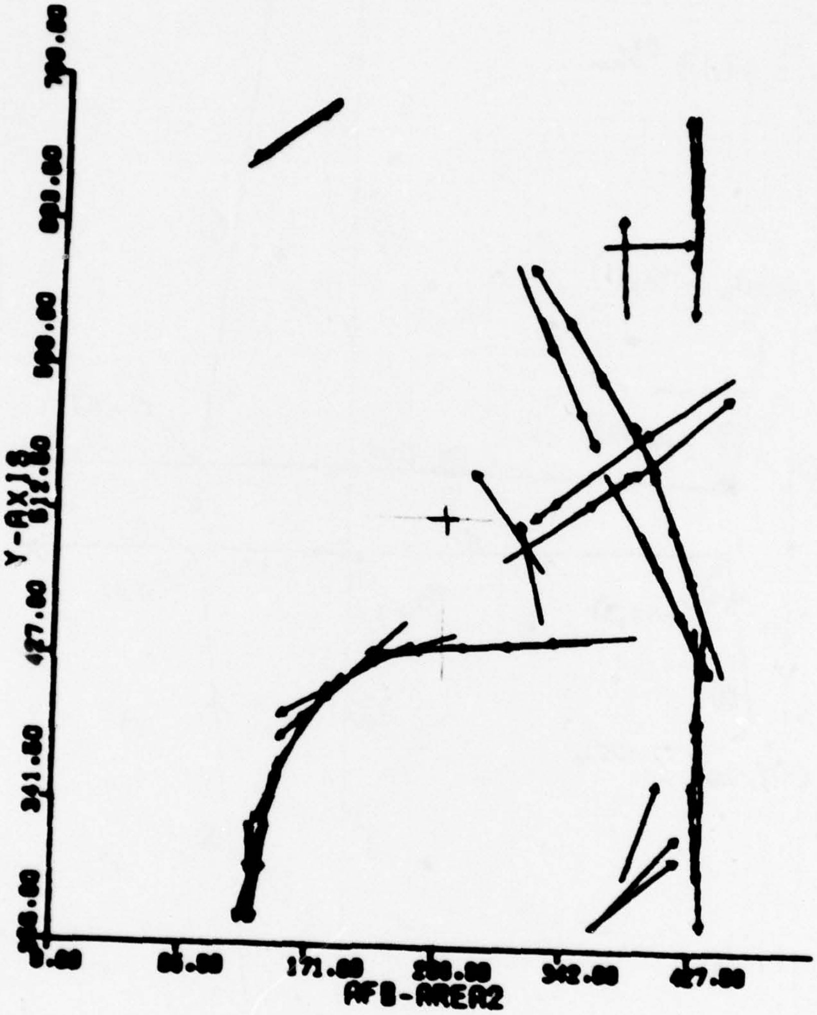


Figure 31. Straight edge segments extracted automatically from a 512 x 512 window of an airfield image.

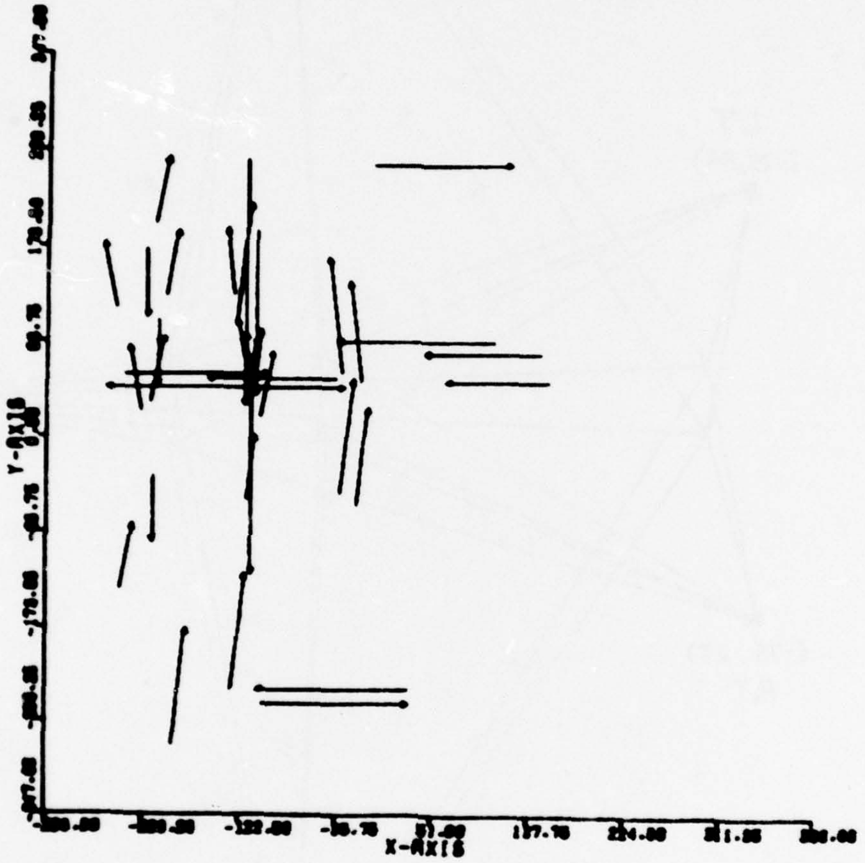


Figure 32. Slice of α-space ($320 \leq \theta \leq 340$) formed by corresponding all pairs of edge segments from Figures 31. and 32. Note cluster for correct α=(330°,xs=-104,ys=43).

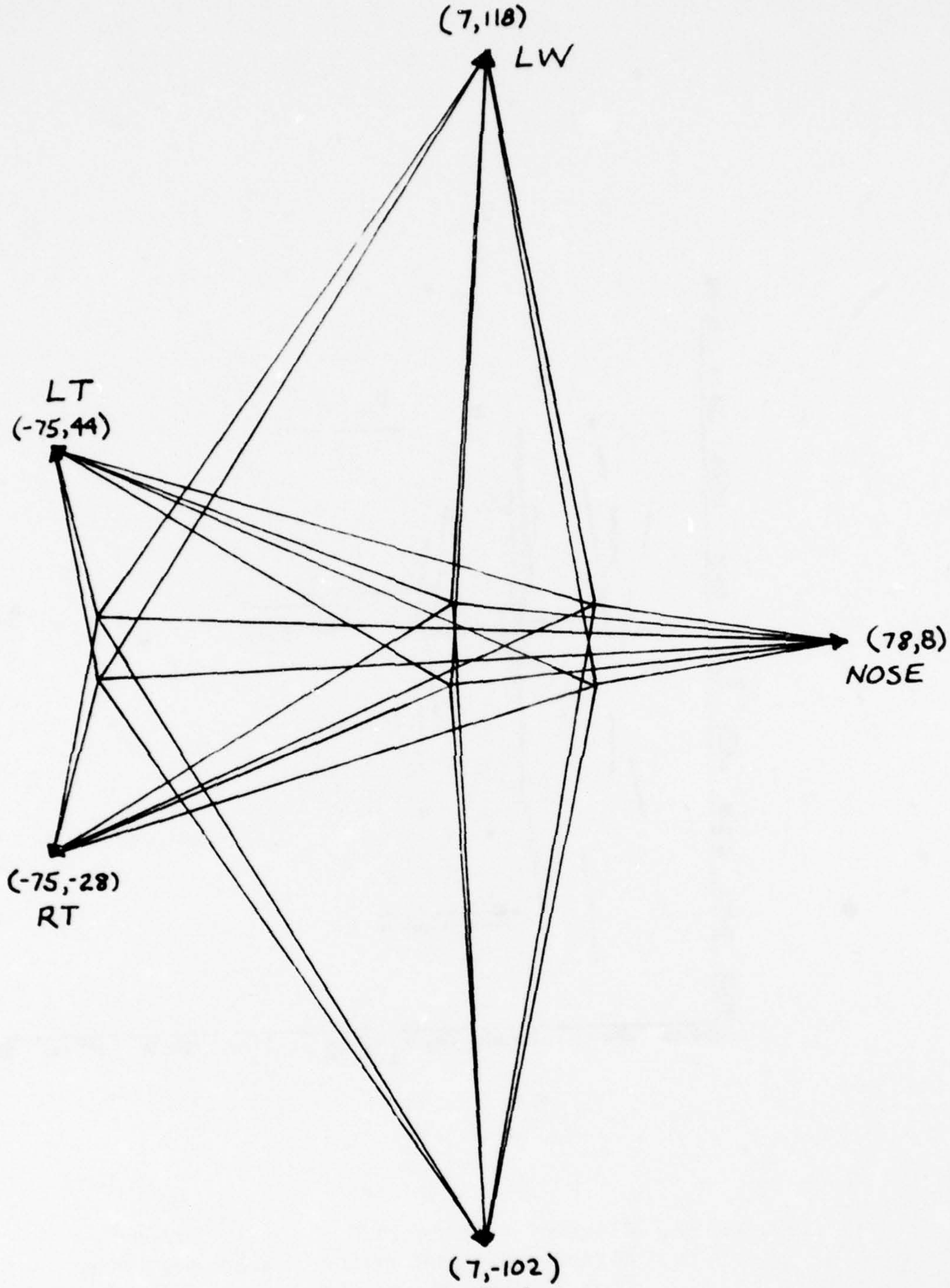


Figure 33. Model of airplane in terms of abstract edges formed from intersections and points of high curvature. (Compare to Figure 30.).

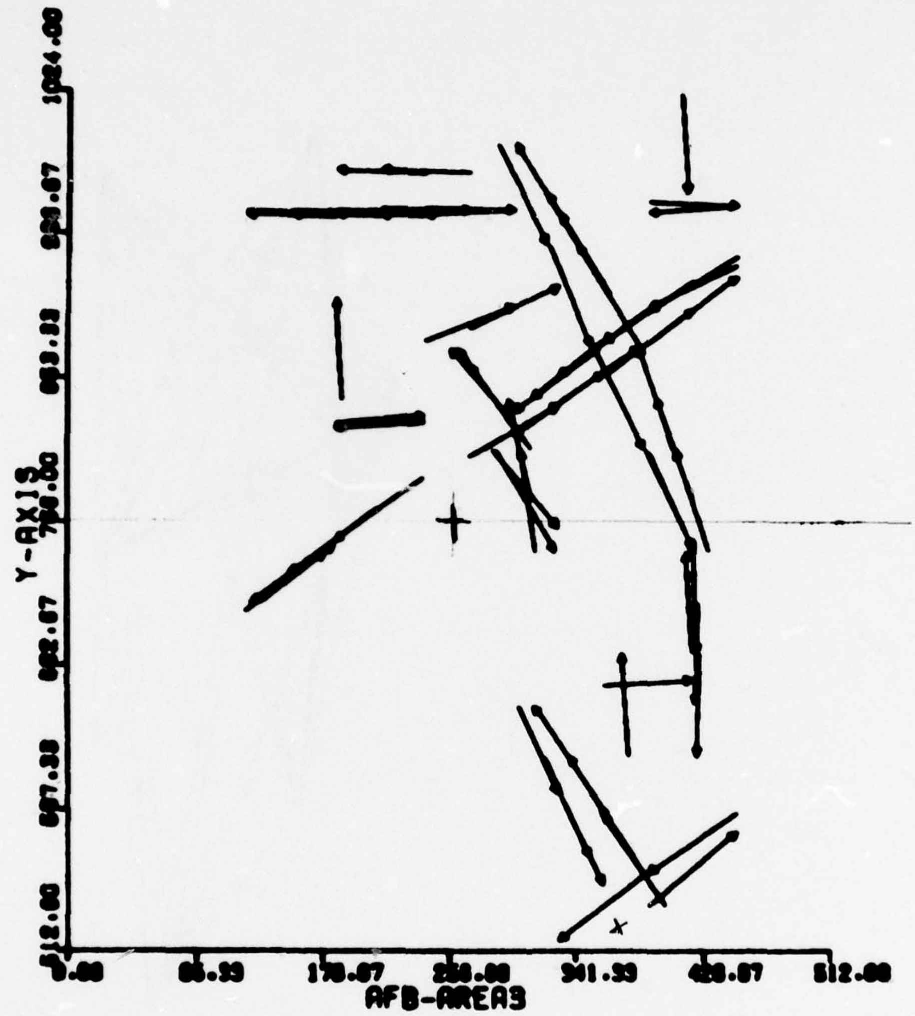


Figure 34. Straight edge segments extracted automatically from a 512 x 512 window of airfield image AFB. (AFB-AREA 3)

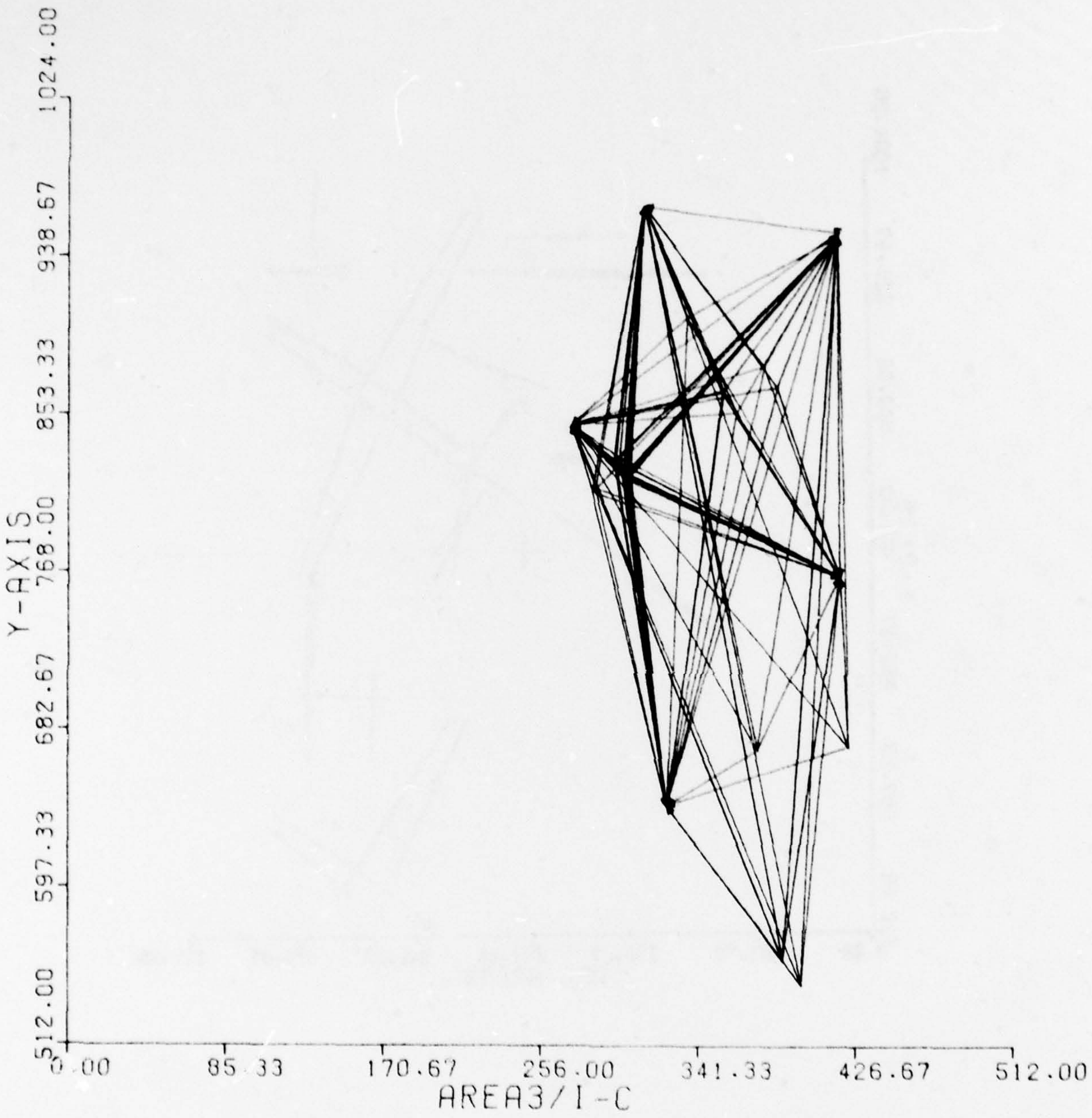


Figure 35. Set of abstract edges formed by connecting intersections to high curvature points automaticallv extracted from image data represented in Figure 34.

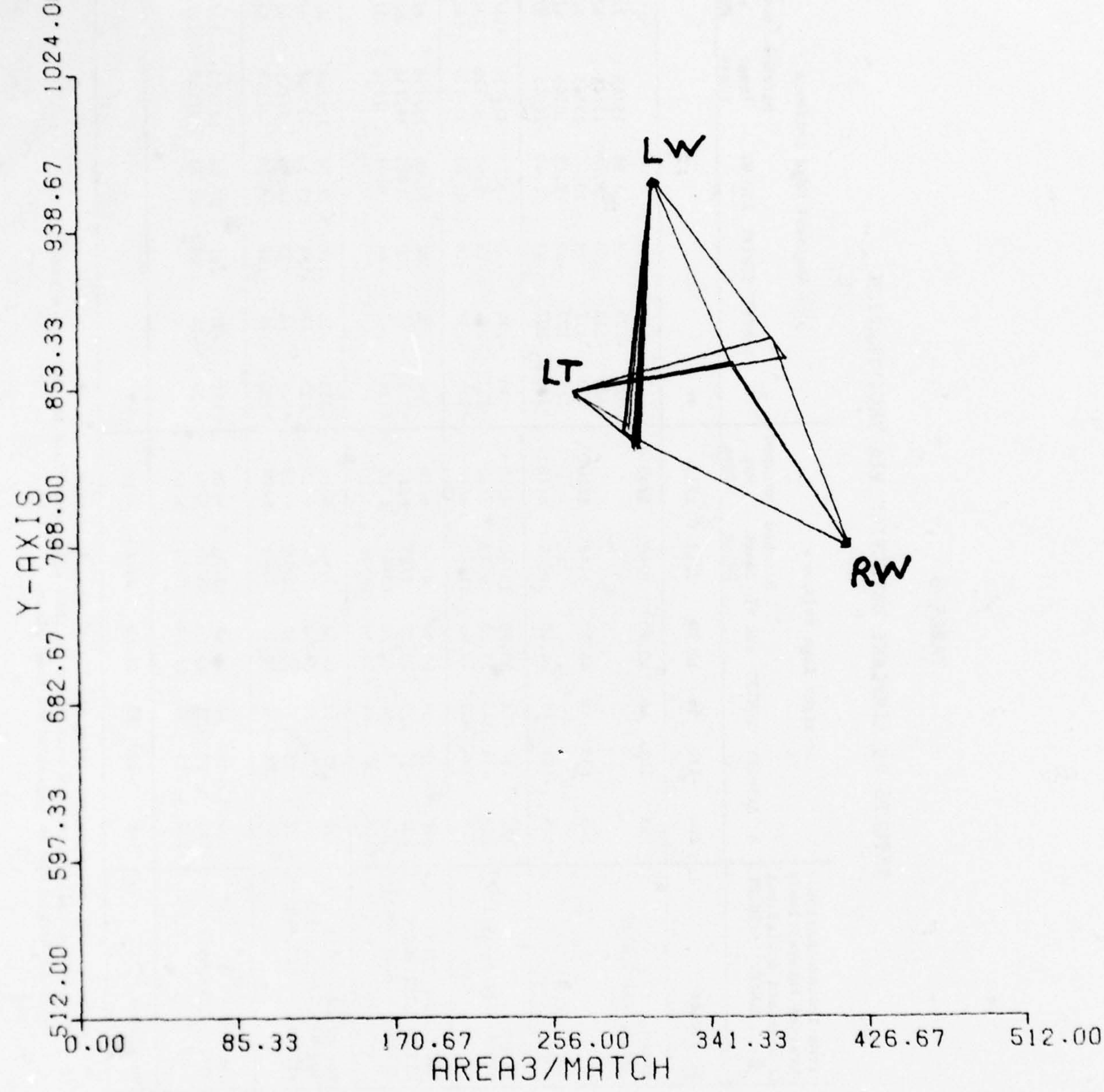


Figure 36. Edge elements of Figure 35. which match edge elements of Figure 34. under automatically derived registration transformation T_α=(θ=330°,xs=-134,ys=-18).

TABLE 3

RESULTS OF AIRPLANE DETECTION VIA REGISTRATION

Center of Sub-Window	True Transformation obtained interactively (± standard deviation)	Straight Edge Evidence						Abstract Edge Evidence					
	θ XSHIFT YSHIFT	θ	XSHIFT	YSHIFT	AVG WT	Matched Image Edges	Matched Map Edges	θ	XSHIFT	YSHIFT	AVG WT	Matched Image Edges	Matched Map Edges
1 (256,256)	No planes	204	-110	56	17.10	3/53	2/10	**					
2 (256,512)	Plane 1 328±1,-107±3,47±2	329	-108	46	42.20	9/40	8/10						
								326	-128	48	15.66	15/45	7/30
								326	-112	50	18.70	13/45	6/30
		145	133	-36	45.10	6/40	6/10	333	-112	28	20.46	13/45	7/30
								148	133	-22	12.13	10/45	4/30
		*328	-107	47	44.40	10/40	8/10	*328	-107	47	21.40	19/45	9/30
3 (256,768)	Plane 2 329±1,-132±2,-19±2	327	-134	-16	35.30	10/40	9/10	330	-134	-18	46.13	31/120	16/30
		57	34	-145	25.10	6/40	5/10	332	-134	-18	35.50	26/120	13/30
		*329	-132	-19	43.60	11/40	9/10	*329	-132	-19	42.03	31/120	16/30
4 (256,1024)	Two planes (2&3) 2) 329±1,1±2,200±4 3) 328±1,-162±2,-84±2	330	-159	-89	52.30	9/48	8/10	326	6	202	22.06	13/156	8/30
		240	-67	178	33.20	7/48	6/10	*329	1	200	21.20	16/156	10/30
		*329	1	200	42.70	10/48	8/10	*328	-162	-84	9.03	5/156	3/30
		*328	-162	-84	54.30	10/48	9/10						
5 (256,1280)	Two planes (3&4) 3) 328±1,-26±3,133±2 4) 330±1,-184±2,-152±2	330	-181	-151	40.10	7/48	8/10	333	-36	129	21.96	12/148	8/30
		56	-110	-24	20.80	7/48	6/10	333	-177	-159	17.36	12/148	6/30
		*328	-26	133	50.80	10/48	9/10	*328	-26	133	14.86	7/148	6/30
		*330	-184	-152	40.40	7/48	8/10	*330	-184	-152	16.60	12/148	6/30
6 (256,1536)	Plane 4 330±1,-55±2,69±2	332	-59	66	63.30	18/70	9/10	332	-61	61	29.06	52/155	10/30
		153	77	-51	40.70	10/70	7/10	*330	-55	69	28.33	57/155	10/30
		*330	-55	69	61.20	18/70	9/10						
7 (256,1792)	No planes	268	-49	12	27.80	4/44	4/10	**					

* - value of avg. match weight using the "true" transformation

** - no points of high curvature were found in this area, therefore no abstract edges could be formed and no registration took place

5. Verification of structures in imagery

In this section it is assumed that enough information has been extracted from an image so that hypotheses about the remaining image content can be made. Our interest is restricted to geometric structures in this report. In the verification of the existence of a particular geometric structure, the rough location and orientation of the structure is known. That is, if the structure exists at all, the model being used should predict approximately where the structure is with respect to previously detected structures, how big it is, what its shape is, etc.. For example, if structures resembling the two wings of an airplane have been detected there are at most two places to search for the tail. Finding a hypothesized structure greatly increases the confidence in the model that generated the hypothesis while failure to detect the predicted structure has just the reverse effect.

Because verification is done with model prediction focused searches can be performed. Not only is the area of imagery to be searched well-confined but there are also tight constraints on shape and orientation. Thus faint or hard to detect features can be found more reliably and more efficiently in the top-down mode than in the bottom-up mode.

Three techniques are described for verification (top-down detection) of geometric structure in imagery. All three techniques operate on edge or gradient information to detect boundary segments of a certain shape. In Section 5.1 "certain shape" is defined by a functional model and the verification paradigm is curve-fitting with χ^2 evaluation of the results. Hough detection of parabolic or circular arcs is discussed in Section 5.2. This approach is essentially template matching where each template is defined by a set of parameters. A practical approach for verifying a fixed but arbitrary curve structure is given in Section 5.3. There a boundary feature is viewed as a set of high gradient points that must be found in the image. This is also a template-matching approach but without parameterization.

5.1 Verification of generic shapes via curve-fitting

The following technique was designed and implemented in 1-D for detection of shape features in waveforms. The system implemented was called WAPSYS, short for waveform parsing system [Stockman 1977]. The treatment will thus be formally given along those lines. A boundary curve in a 2-D image is basically a 1-D entity and so the 1-D method can be converted to the 2-D case. Methods for conversion and the problems encountered are discussed at the end of this section.

5.1.1 Extracting primitives by curve-fitting

We define a waveform as a finite function $W=\{(x_i,y_i)\},i=1,N$; i.e. there are a finite number N of pairs (x_i,y_i) and no two pairs have the same first element. Since x is a time or distance variable, there is a linear order imposed on the set of points, i.e. x_i is before x_{i+1} and after x_{i-1}. The task is to verify if some subset (subinterval) of points $\{(x_j=a,y_j)\ldots(x_k=b,y_k)\}$ has shape M. M is a shape primitive or morph. The set of points on which the search for M is constrained is called the constraint interval and is indicated as interval $[\ell,r]$. The subinterval on which morph M is detected, if it is detected, is called the match interval. The detector may, in fact, identify no occurrence, one occurence, or many occurrences of the morph $<M,[a_j,b_j],e_j,p_j>$ existing on the constraint interval. P_j is the parameterization of the morph M and e_j is an evaluation of its merit or certainty. The morph M is specified to the detector by a syntactic name and semantic constraints C which must be satisfied by parameterization P. M might be formally defined as a functional form $y_m=y_m(x)=f(x)$ to be fit to the data $y(x), x\varepsilon[a,b]$ under set of constraints C.

For example, the 'cap' morph of Figure 37 could be defined as $y_m(x)=p_2x^2 + p_1x+p_0$, subject to the constraints that $y_m(a)=y_m(b), c_1 < p_2 < c_2 < 0$ and $c_3 \leq b-a < c_4$. The parameterization $P=\{P_0,P_1,P_2\}$ could naturally be determined by least squares fitting $y_m(x)$ to $y(x)$ over $x\varepsilon[a,b]$. Figure 37 shows 5 of the morphs used extensively in a pulse wave application detailed in [Stockman 1977]. All 5 morph definitions imply least squares error estimation of 2 free parameters. Under the assumption of Gaussian noise distributed as $N(0,\sigma^2)$ the variable

$$s = \sum_{x=a}^{b} (y_m(x)-y(x))^2)/\sigma^2 \text{ is } \chi^2 \text{ distributed with}$$

b-a-1 degrees of freedom if the values of $y(x)$ are interpreted as realizations of $y_m(x)$ plus noise. By defining e as the probability that $\chi^2(z,b-a-1)>s$, the merit or certainty of the fit (or morph hypothesis) is nicely bounded in [0,1] and may indeed be interpreted as a probability.

Classically, all n given points $(x_i, y_i), x_i \varepsilon [\ell, r]$ must fit and the noise σ^2 is known. This allows variation only in model. The approach in polynomial fitting is to vary upward the degree m of the model polynomial

$$y = y_m(x) = \sum_{i=0}^{m} p_i x^i$$

until the hypothesis H(m), that this model generated the data, can be accepted at a given confidence and the hypothesis H(m+1) does not increase significantly this confidence of fit [Ralston 1965]. The approach here is to keep the polynomial form $y_m(x)$ fixed in form and to vary the subinterval $[a,b] \subseteq [\ell, r]$ to find the best fit(s). The reasons for doing this is that it is desired that the data be represented by morphs of confined geometric shape whose parameters might have strong interpretation in the problem domain. For instance, in searching for the nose of a plane it is quite significant if a smooth symmetric parabolic shape is found as opposed to two steeply sloping straight lines forming a sharp peak.

Figure 38 shows some pulse wave data that was fitted with models from Figure 37. The morphs UP,MPS,MNG,LN, and HOR are defined and detected by using constraints on the parameters of the straight line model. The CAP morph and RSH morph are instances of the cap and right shoulder of Figure 37. Constraining the juxta-position of these morphs is in the domain of syntax which was discussed in Section 3.

5.1.2 A curve-fitting detector

To find the optimum fit(s) of a given morph on some subinterval [a,b] of constraint interval $[\ell, r]$, a large number of fits might have to be tried. If the bounds c_3 and c_4 of b-a are stringent and the interval $[\ell, r]$ is not big, the entire interval

$[\ell,r]$ can be scanned for all subintervals [a,b] satisfying $c_3 \leq b-a \leq c_4$. Such exhaustive scanning can be enforced by WAPSYS and guarantees that the optimal fit under the given constraints is found. If the morph width b-a can vary widely or if the interval $[\ell,r]$ is big, optimality can be sacrificed for efficiency through a heuristic search alternative. Details of the heuristic search alternative can be found in [Stockman 1977] or [Stockman 1978] and are ignored here because it is assumed that for verification the search interval $[\ell,r]$ will be well enough constrained for exhaustive search. Exhaustive search is faster under strong constraints, partly because recursively updateable curve fitting can be done.

With exhaustive search a fair number of fits may be made and several of them may yield high χ^2 values. All fits (detections) above a certain threshold are saved in a ranked list while those below threshold are discarded. The list of qualifying detections is passed back to the syntax/semantics module for further analysis. Since the size of a morph is constrained over a flexible range it will often happen that one detection will overlap another or be entirely contained in another. For this reason pruning is done on the list of detections so that redundant detections are discarded. Let each interval of data [a,b] fit by the detector in search of morph M be called state [a,b]. The conditions for suppressing one detection in the presence of another are set forth as follows.

Definition 5.1

Let [a,b] and [c,d] be two intervals of the real line and let $s([x_1,x_2])=x_2-x_1$ be the length measure for interval $[x_1,x_2]$. The overlap or correlation between intervals [a,b] and [c,d] is defined as $\text{OVRLAP}([a,b],[c,d])=2s([a,b]\cap[c,d])/(s([a,b])+s([c,d]))$.

Definition 5.2

Let interval $[L_i,R_i]$ be fit with quality E_i and $[L_j,R_j]$ be fit with quality E_j in two states of the curve fitting. State $[L_i,R_i]$ is said to dominate state $[L_j,R_j]$ if and only if

a) $E_i \geq E_j$

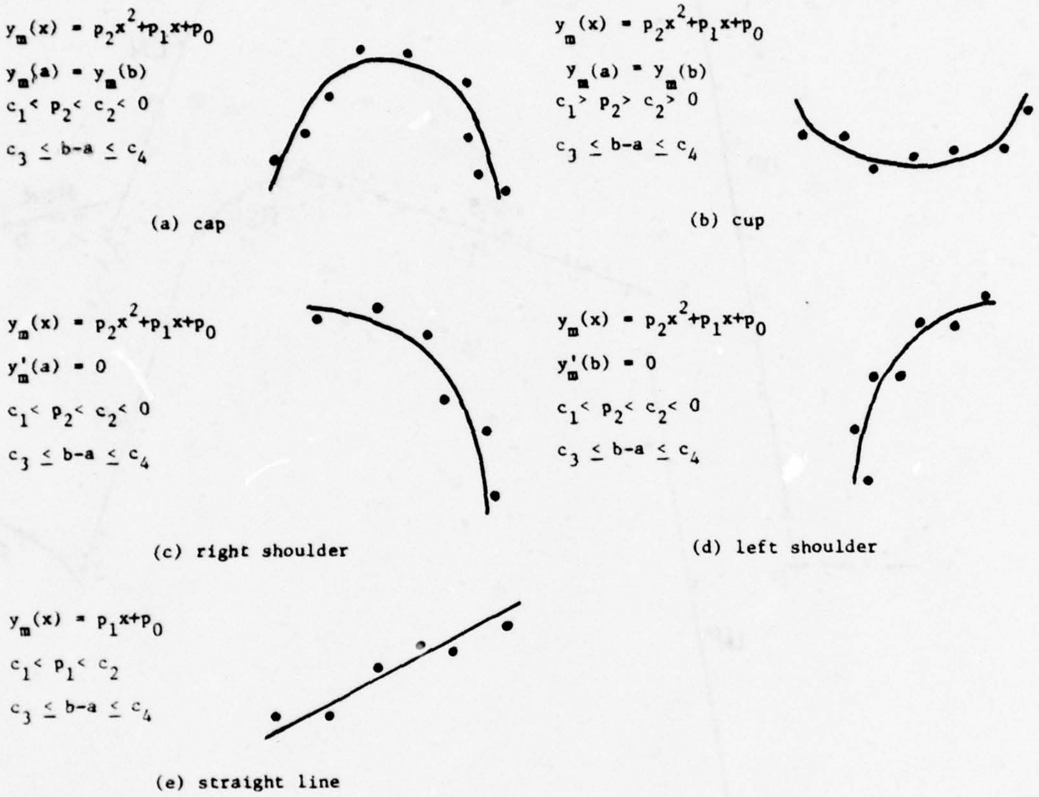
and b) $[L_j,R_j] \subseteq [L_i,R_i]$ or

$\mathrm{OVRLAP}([L_i,R_i],[L_j,R_j]) > t$ and

$s([L_j,R_j]) \leq s([L_i,R_i])$.

Roughly speaking, fit i dominates fit j if it has at least as good a quality and covers at least as many points in the same general region of data.

The 5 morphs described in Figure 37 were implemented as FORTRAN subroutines for use in several domains. The subroutines CAPFIT,LINFIT,RSHFIT,LSHFIT, compute the fit parameters for a given subinterval of data and also return the sum of the squared errors. A driver program (FITTER) controls the choice of subintervals to be fit and interprets the quality of fits by calling routine QCHISQ to make the chi-quared comparison of fit error to noise. The set of fit constraints $C=(c_1,c_2,c_3,c_4)$ is gotten by FITTER via a call to GETCST, which gets them from tabularized input from the user. FITTER returns when the search is complete. An entry point NXTFIT is then available to get successively worse fits from the list of FITTER detections.



$y_m(x) = p_2x^2+p_1x+p_0$

$y_m(a) = y_m(b)$

$c_1 < p_2 < c_2 < 0$

$c_3 \le b-a \le c_4$

(a) cap

$y_m(x) = p_2x^2+p_1x+p_0$

$y_m(a) = y_m(b)$

$c_1 > p_2 > c_2 > 0$

$c_3 \le b-a \le c_4$

(b) cup

$y_m(x) = p_2x^2+p_1x+p_0$

$y_m'(a) = 0$

$c_1 < p_2 < c_2 < 0$

$c_3 \le b-a \le c_4$

(c) right shoulder

$y_m(x) = p_2x^2+p_1x+p_0$

$y_m'(b) = 0$

$c_1 < p_2 < c_2 < 0$

$c_3 \le b-a \le c_4$

(d) left shoulder

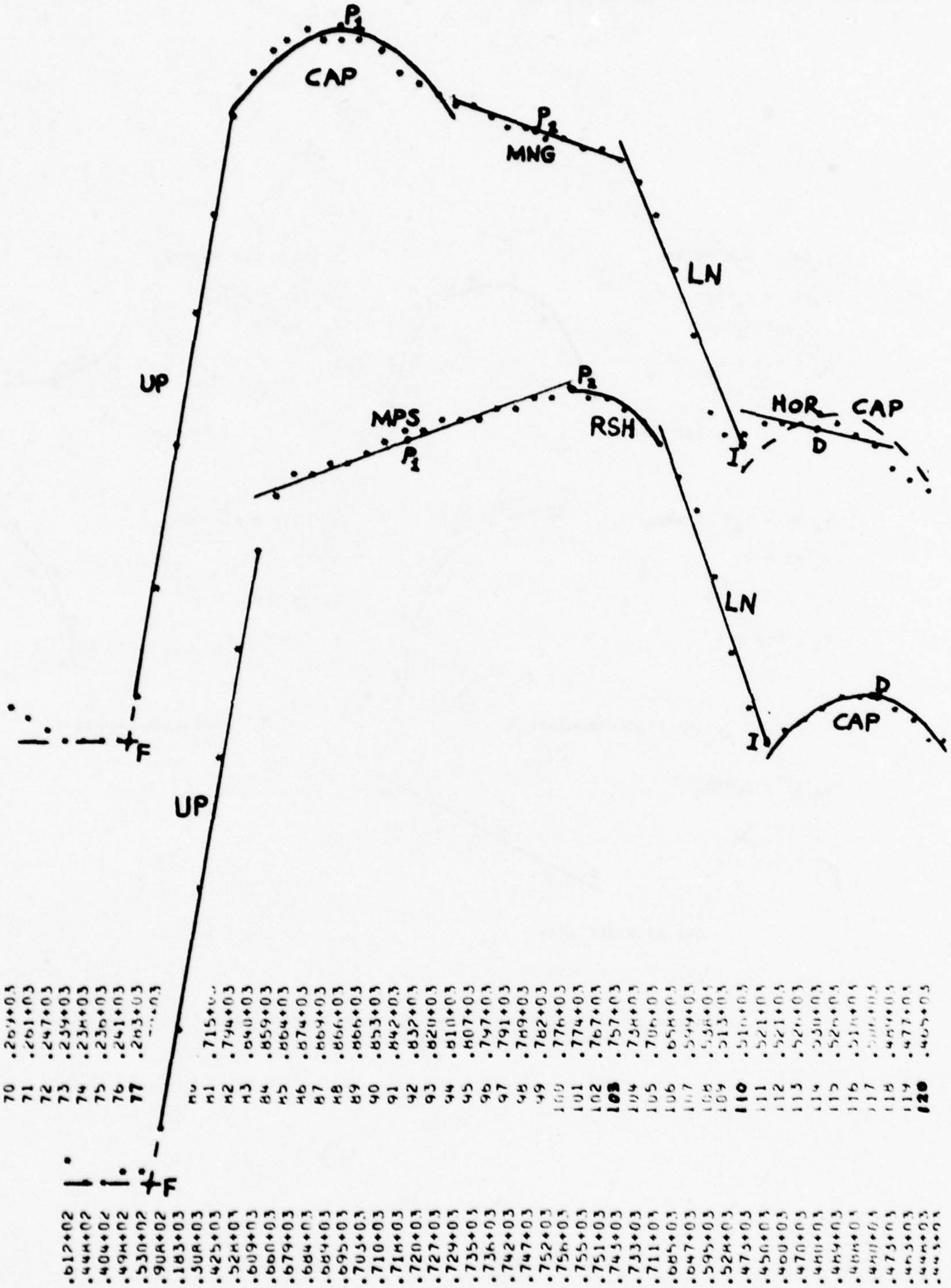
$y_m(x) = p_1x+p_0$

$c_1 < p_1 < c_2$

$c_3 \le b-a \le c_4$

(e) straight line

Figure 37. Five morphs defined by constrained fits of model $y_m(x)$ to data $y(x)$, $x\varepsilon[a,b]\subseteq[\ell,r]$.



```
 70  .269+03
 71  .261+03
 72  .247+03
 73  .239+03
 74  .238+03
 75  .236+03
 76  .241+03
 77  .263+03
     [illegible]
 80  [illegible]
 81  .715+03
 82  .794+03
 83  .840+03
 84  .859+03
 85  .864+03
 86  .874+03
 87  .869+03
 88  .866+03
 89  .866+03
 90  .853+03
 91  .842+03
 92  .832+03
 93  .820+03
 94  .810+03
 95  .807+03
 96  .797+03
 97  .791+03
 98  .789+03
 99  .782+03
100  .776+03
101  .774+03
102  .767+03
103  .757+03
104  .738+03
105  .706+03
106  .658+03
107  .599+03
108  .558+03
109  .513+03
110  .510+03
111  .521+03
112  .521+03
113  .526+03
114  .530+03
115  .526+03
116  .518+03
117  .500+03
118  .489+03
119  .477+03
120  .465+03
```

```
158  .612+02
159  .448+02
160  .404+02
161  .498+02
162  .530+02
163  .908+02
164  .183+03
165  .308+03
166  .425+03
167  .528+03
168  .609+03
169  .660+03
170  .679+03
171  .684+03
172  .689+03
173  .695+03
174  .703+03
175  .710+03
176  .718+03
177  .720+03
178  .727+03
179  .729+03
180  .735+03
181  .736+03
182  .742+03
183  .747+03
184  .752+03
185  .756+03
186  .755+03
187  .751+03
188  .743+03
189  .733+03
190  .711+03
191  .685+03
192  .647+03
193  .595+03
194  .528+03
195  .473+03
196  .450+03
197  .460+03
198  .470+03
199  .480+03
200  .484+03
201  .488+03
202  .488+03
203  .473+03
204  .483+03
205  .445+03
```

Figure 38. Carotid pulse wave sample segmented into primitive shape features.

5.1.3 Training and testing curve fitting detectors

While general morphs such as the 'cap' or the 'line' might be applicable in several different problem domains, it is unlikely that the constraints would be defined in the same way in each domain. These constraints could be defined from a priori considerations or could be "learned" from data samples. WAPSYS can be used to define morph constraints by fitting training data under no constraints and recording the parameterizations that result. The set of parameterizations for each named morph can then be converted to constraints for use in automatic analysis.

WAPSYS was used to learn the constraints necessary for automatic parsing of carotid pulse waves. 14 morphs were identified in this application and all were expressible in terms of constraints on generic features as shown in Figure 27 Eight hours was spent by the author examining print plots of 20 sample pulse waves. Aided only by a ruler, all the data was segmented, producing for each sample a list of triples $<M_i,a_i,b_i>$ where M_i was a 3 character morph name and $[a_i,b_i]$ was the interval of data on which the morph was declared to exist. 459 morphs were identified in this manner for the 20 waveforms. A training routine was written which called the fitting routine appropriate for each morph specified and forced it to fit the specified data interval. The fit was forced in the sense that the noise tolerance was varied upward until either a limit was reached or a fit of quality above 0.5 was achieved. (The noise limit was useful for detecting human errors in creation of the training items--7 or 8 errors were detected in this manner). The parameters of a successful fit were contributed to a running statistical summary which was the final output of training.

The statistical summary contained, for each morph, the lowest, highest, and mean value for each of 3 variables and the standard deviation. The variables were interval width b-a, noise σ^2, and curvature for parabolic morphs or slope for linear ones. 3 iterations of training were required to remove all human errors in the 459 training items. The summary values from training were then used to construct a table of constraints to be used in automatic detection by the FITTER algorithm. A testing module was written for WAPSYS to drive the FITTER algorithm in an attempt to automatically identify the same morphs that the human user did in constructing the list of training items. The testing routine reads each training item $<M_i,a_i,b_i>$, computes an enlarged constraint interval $[\ell_j,r_j]$ from $[a_i,b_i]$ and calls the FITTER to detect morph M_i on the interval $[\ell_i,r_i]$. The list of successful fits (if any) is then scanned to find that fit which correlates best with training item. A report of the fits achieved for each training item is given as a final summary of how well the morphs detected correlated with the training items. The constraint interval for each training item is computed as

$$\ell_i = \max(1, a_i - 2(b_i - a_i + 1)z_1)$$

$$r_i = \min(n, b_i + 2(b_i - a_i + 1)z_2)$$

where n is the total number of points in the waveform and z_1 and z_2 are independent samples from the unit normal distribution N(0,1). The correlation between two fits was given in Definition 1. Since many fits can be detected in testing, the best correlating fit is used to contribute to the correlation summary. Results of testing should indicate whether or not the models used for detection can or cannot succeed in a given application. Failure in the training and testing cycle indicates the need to define a different set of primitives. In the carotid pulse wave application the results of training and testing were very good. These results and the results of further analysis of a larger set of waveforms is discussed next.

5.1.4 Results and discussion of carotid pulse wave analysis

Table 4 shows the performance of the fitting algorithm in testing on the 20 samples used for training. Almost 9000 fits were tried and 3600 detections were made. In 186 of 449 cases the human selected fit interval was detected automatically. The best correlating match averaged 0.944 meaning that there was little disagreement between man and machine. A large number of the fits tried and successfully made were due to straight line morphs. Since a straight line is uniform any subinterval of a large segment would have the same parameters and hence be detected. Search for parabolic morphs was more focused and fewer detections were made.

Since the results of testing on the training set were quite satisfactory, the tuned WAPSYS was then applied to 158 waveforms that were not previously considered. Syntax and semantics coded in a PRR were also employed. Performance on the test samples was not as good as on the training samples. There were unforseen structural variations and there was fluctuation in the noise.

As is commonly reported, the noise characteristics of the data were not those assumed by the model! Not only did the noise vary from one waveform to the next but it also varied within the same waveform. A low noise allowance would cause morphs to be missed in noisy waveforms while a high allowance would cause sloppy detections to dominate fine ones in clean data. Another problem existing is model bias where, for example, the user creates a linear morph to visually fit logarithmic data. Clearly the model noise will not be Gaussian.

Figure 38 shows a syntactic labeling of 2 consecutive cycles of the same pulse wave. Note the competetion between the straight line morph HOR and the parabolic CAP for representation of the data in the interval [110,120]. Note also that interval [77,103] is represented by a cap and a line while interval [168,190] is represented by a line and a right shoulder. This is somewhat alarming, shouldn't the human body be behaving the same way during both these intervals?

Table 5 supports the claim that higher level knowledge can be used to make lower level detection more efficient. In comparing total analysis (detection + syntax + semantics) of 158 waveforms with detectors testing (no syntax or semantics) on 20 different training waveforms it is apparent that the interaction of higher level processes in the detection process can pay for itself in terms of computation times. The syntactic and semantic modules of WAPSYS were as described in Section 3.

Performance of the detection module of WAPSYS was excellent on the 20 training sampies. Performance degraded on the test set: some morphs on some of the waveforms were badly fit and some of the waveforms were not parsed. In many of these cases failure was the desired outcome. Apparently our linguistic model for one region of the data was inadequate and no easy fixes are evident.

Due to infrequent failures in the modeling and computer times longer than real time a system such as WAPSYS may be inadequate for monitoring of sensors but may be quite good for research or other interactive off-line operation. Graphics such as in Figure 38 are easily available to the user and the analysis readily communicated due to the linguistic description obtained. Approval of the automatic analysis saves the user time. Analyses unacceptable to the user can be easily adjusted due to the common linguistic model between man and machine.

TABLE 4

PERFORMANCE OF FITTER ON 20 PULSE WAVES TRAINING SAMPLES

number of morphs	449
total number of fits tried	8773
total number of detections	3576
total number of perfect matches	186
average best match	0.944
number of morphs detected	449
number of morphs correctly detected	449
total seconds of computer time	211+

+ Univac 1108 computer used

TABLE 5

WAPSYS EXECUTION DATA FOR SEVERAL ANALYSES OF 60 CYCLES OF PULSE WAVES*

	Total Fits	Total Detections	Univac 1108 Sup Time In Seconds Per 60 Cycles	Number Of Samples Based On
detector training	449	449	32	20 training
detector testing	8800	3600	210	20 training
total analysis*	6800	2600	190	158 testing

*Primitive detection (no syntax or *semantics*) for 20 training samples (60cycles) is compared with *total* analysis (detection + syntax + semantics) of 158 testing samples. Execution time is normalized to 60 cycles of data for comparison. The "cycle" has a constant syntactic character but can vary widely in number of points.

5.1.5 Performance of curve fitting in 2-D

Extraction of primitives via curve fitting in 2-D was briefly discussed in Section 3.4 with respect to the detection of an airplane wing. A more detailed example is given here which shows how the 1-D curve fitters of WAPSYS were gotten to work on boundary curves from 2-D imagery.

Imagery containing a parabolic boundary curve is shown in Figure 39. The hypothesis to be verified is that the curve runs from point (58,49) to point (30,49) of the image. In this case the hypothesis was generated by the author and not by a PRR model as would be the case with totally automatic processing. The hypothesis is used to establish a new coordinate system as shown in Figure 39. The image gradient is then sampled along profiles generated along constant x coordinates in the new coordinate system. From each profile the highest above threshold gradient magnitude is selected as the best point on the possible boundary curve and the entire point set $W=\{(x_i,y_i)\}i=1,N$ is passed to the curve-fitter. Figure 40 shows the points selected on the gradient profile from the image of Figure 39. Since only one point (x_i,y_i) is selected on the i-th profile, W is itself a function as is a waveform. Exhaustive search of W for a parabolic CAP (Figure 37) is shown in Figure 41. All 45 possible intervals of 20 or more points are tried. Most fits [a,b] are failures ($\chi^2=0$) even though the points do lie along a parabola because of the constraint that $Y_m(a)=Y_m(b)$. A good fit is achieved on [3,22] and is later suppressed by a dominating fit on [1,27]. The value of that fit is 1.00 via comparison with a given noise allowance. The fitted curve runs from points (58,54) to (32,54) in the image and correlates 0.544 with the original hypothesis vector (58,49)→(30,49).

XAPPIC.GAFB   XAPIN 24,246,83,305

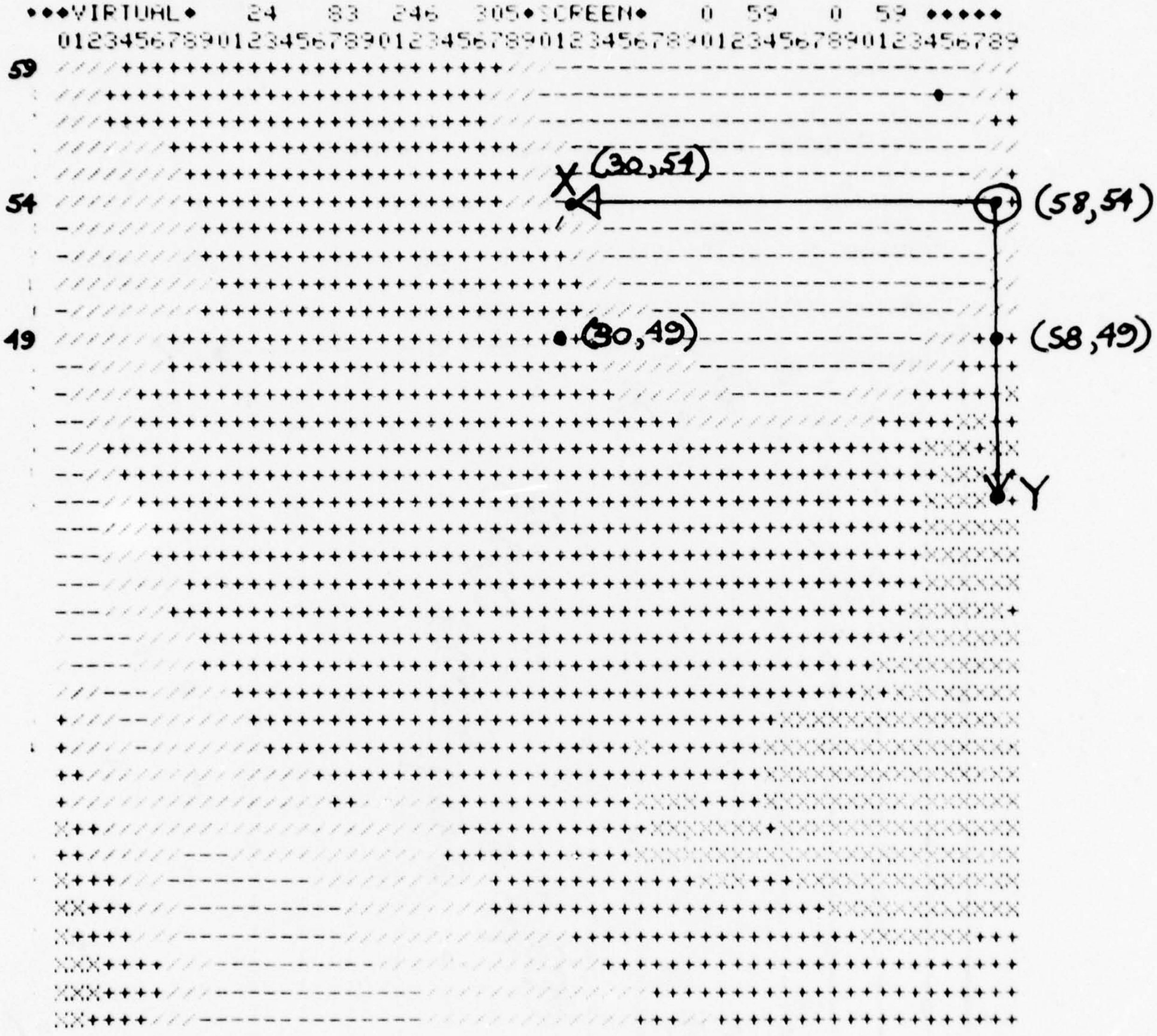
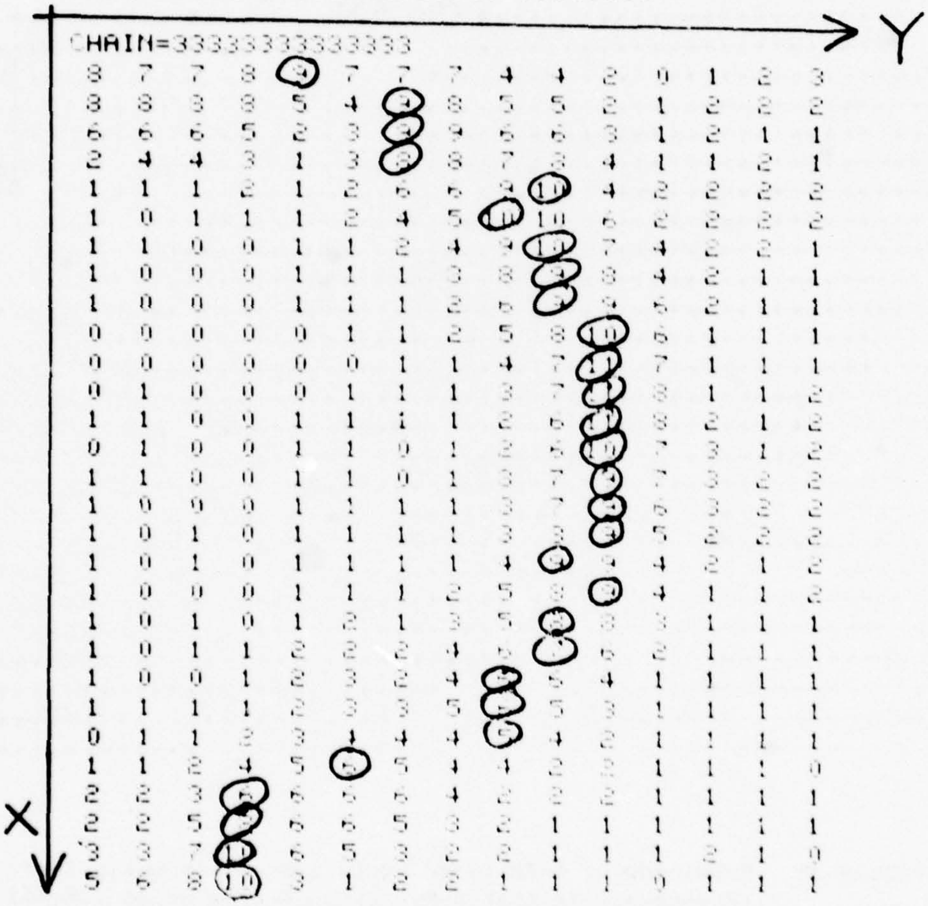


Figure 39. Subwindow of GAFB image containing a parabolic curve. (Hypothesis is that curve extends from point (58,49) to point (30,49) and has negative curvature.)

```
[illegible]

APIN 24,246,83,305
UBIMAGE      24   246    83   305 READ O.K.

TABLE

RAD

APSYS

CAN  1,58.0,49.0,30.0,49.0,50.0,0.8,
CAN:INDEXP,HXB,HYB,HXE,HYE,HSIZE,OTHRES=
   1
 .580+02  .490+02  .300+02  .490+02
 .500+02  .800+00
CAN:XC,YC,XX,XY,YX,YY=   58   56   30   56   58   42

HAIN=555555555555555555555555
```



```
HAIN=33333333333333
 8  7  7  8  9  7  7  7  4  4  2  0  1  2  3
 8  8  8  8  5  4  9  8  5  5  2  1  2  2  2
 6  6  6  5  2  3  9  9  6  6  3  1  2  1  1
 2  4  4  3  1  3  8  8  7  8  4  1  1  2  1
 1  1  1  2  1  2  6  6  9 10  4  2  2  2  2
 1  0  1  1  1  2  4  5 10 10  6  2  2  2  2
 1  1  0  0  1  1  2  4  9 10  7  4  2  2  1
 1  0  0  0  1  1  1  3  8  9  8  4  2  1  1
 1  0  0  0  1  1  1  2  6  9  9  6  2  1  1
 0  0  0  0  0  1  1  2  5  8 10  6  1  1  1
 0  0  0  0  0  0  1  1  4  7 10  7  1  1  1
 0  1  0  0  0  1  1  1  3  7 10  8  1  1  2
 1  1  0  0  1  1  1  1  2  6 10  8  2  1  2
 0  1  0  0  1  1  0  1  2  5 10  7  3  1  2
 1  1  0  0  1  1  0  1  3  7  9  7  3  2  2
 1  0  0  0  1  1  1  1  3  7  9  6  3  2  2
 1  0  1  0  1  1  1  1  3  8  9  5  2  2  2
 1  0  1  0  1  1  1  1  4  9  9  4  2  1  2
 1  0  0  0  1  1  1  2  5  8  9  4  1  1  2
 1  0  1  0  1  2  1  3  5  8  8  3  1  1  1
 1  0  1  1  2  2  2  4  6  8  6  2  1  1  1
 1  0  0  1  2  3  2  4  6  6  4  1  1  1  1
 1  1  1  1  2  3  3  5  6  5  3  1  1  1  1
 0  1  1  2  3  4  4  4  6  4  2  1  1  1  1
 1  1  2  4  5  6  5  4  4  2  2  1  1  1  0
 2  2  3  6  6  6  6  4  2  2  2  1  1  1  1
 2  2  5  8  6  6  5  3  2  1  1  1  1  1  1
 3  3  7 10  6  5  3  2  2  1  1  1  2  1  0
 5  6  8 10  3  1  2  2  1  1  1  0  1  1  1
```

Figure 40. Profiles of the gradient sampled with respect to the xy-coordinate axes labeled in Figure 39. (Points of peak gradient are input to curve fitter.)

```
              a_i  b_i   e_i         curvature
FITTER: (     1.   20.  .0000      ) A= .1304+00
FITTER: (     2.   21.  .0000      ) A= .9641-01
FITTER: (     3.   22.  .1000+00)  A= .5386-01
FITTER: (     4.   23.  .1000+01)  A= .4016-02
FITTER: (     5.   24.  .2500+00)  A=-.5357-01
FITTER: (     6.   25.  .0000      ) A=-.1373+00
FITTER: (     7.   26.  .0000      ) A=-.2733+00
FITTER: (     8.   27.  .0000      ) A=-.3926+00
FITTER: (     9.   28.  .0000      ) A=-.5064+00
FITTER: (    10.   29.  .0000      ) A=-.6132+00
FITTER: (     1.   21.  .0000      ) A= .1056+00
FITTER: (     2.   22.  .5000-01)  A= .6345-01
FITTER: (     3.   23.  .8000+00)  A= .2795-01
FITTER: (     4.   24.  .1000+01)  A=-.1343-01
FITTER: (     5.   25.  .0000      ) A=-.1070+00
FITTER: (     6.   26.  .0000      ) A=-.2054+00
FITTER: (     7.   27.  .0000      ) A=-.3130+00
FITTER: (     8.   28.  .0000      ) A=-.4062+00
FITTER: (     9.   29.  .0000      ) A=-.4944+00
FITTER: (     1.   22.  .0000      ) A= .7628-01
FITTER: (     2.   23.  .3500+00)  A= .4028-01
FITTER: (     3.   24.  .1000+01)  A= .1039-01
FITTER: (     4.   25.  .3000+00)  A=-.6172-01
FITTER: (     5.   26.  .0000      ) A=-.1646+00
FITTER: (     6.   27.  .0000      ) A=-.2431+00
FITTER: (     7.   28.  .0000      ) A=-.3290+00
FITTER: (     8.   29.  .0000      ) A=-.4024+00
FITTER: (     1.   23.  .0000      ) A= .5488-01
FITTER: (     2.   24.  .8500+00)  A= .2391-01
FITTER: (     3.   25.  .1000+01)  A=-.3221-01
FITTER: (     4.   26.  .0000      ) A=-.1136+00
FITTER: (     5.   27.  .0000      ) A=-.1980+00
FITTER: (     6.   28.  .0000      ) A=-.2611+00
FITTER: (     7.   29.  .0000      ) A=-.3303+00
FITTER: (     1.   24.  .1500+00)  A= .3914-01
FITTER: (     2.   25.  .1009+01)  A=-.1313-01
FITTER: (     3.   26.  .1000+00)  A=-.7808-01
FITTER: (     4.   27.  .0000      ) A=- 1456+00
FITTER: (     5.   28.  .0000      ) A=- 2154+00
FITTER: (     6.   29.  .0000      ) A=-.2565+00
FITTER: (     1.   25.  .9500+00)  A= .6299-02
FITTER: (     2.   26.  .1000+01)  A=-.5323-01
FITTER: (     3.   27.  .0000      ) A=-.1077+00
FITTER: (     4.   28.  .0000      ) A=-.1641+00
FITTER: (     5.   29.  .0000      ) A=-.2224+00
FITTER: (     1.   26.  .1000+01)  A=-.2933-01
FITTER: (     2.   27.  .3500+00)  A=-.3005-01
FITTER: (     3.   28.  .0000      ) A=-.1260+00
FITTER: (     4.   29.  .0000      ) A=-.1735+00
FITTER: (     1.   27.  .1000+01)  A=-.3398-01
FITTER: (     2.   28.  .0000      ) A=-.9746-01
FITTER: (     3.   29.  .0000      ) A=-.1365+00
FITTER: (     1.   28.  .9500+00)  A=-.7070-01
FITTER: (     2.   29.  .0000      ) A=-.1082+00
FITTER: (     1.   29.  .2000+00)  A=-.8167-01

FITTER:    45 FITS TRIED BY CAP
            1 HAD QUALITY ABOVE THRESHOLD    .800+00
               2 MAX NUM STATES FILED
SCAN2:X1,Y1,X2,Y2,A,B= .000     .223+01 .260+02 .223+01 -.540-01 .140+01
VERLIN:A,D,F,COSINA=
  .580+02   .490+02   .300+02   .490+02   .500+02
  .580+02   .538+02   .320+02   .538+02   .100+01
  .544+00   .100+01        (58,54)   (32,54)
Q,XB,YB,XE,YE= .544+00 .580+02 .538+02 .320+02 .538+02
```

Figure 41.

Exhaustive search for best fitting parabola in subintervals of 29 data points circled in Figure 40. (Best detection is $y_m(x)=-0.054x^2+1.4x+c$ on interval $x \in [1,27]$ with $\chi^2=1.0$. Curve runs from point (58,54) to point (32,54) in image coordinates of Figure 39. and has verification value 0.544 w.r.t. hypothesis (58,49)→(30,49).)

$y = -.054x^2 + 1.4x + C$

assume $C = 2$

A PRR model was actually used to generate hypotheses for CAP detection in looking for airplane wings as discussed in Section 3.4. That experiment exposed a major flaw in the point sampling technique. Because only one point was selected on a profile the globally best set of points was usually missed. This is evident in Figure 40 where a smoother set of points can be chosen by man once the global structure is seen. In the airplane wing this phenomena was particularly acute where the parabolic tip joined the straight side.

Clearly a truly 2-D approach is required but there are more nuissances in 2-D. Cooper [1976] discusses true 2-D detection and isolates 3 problem steps.

P1) Given a fit on n points in the plane, the algorithm must choose the next point for possible extension of the fitted curve. (This is not in general easy to do and there is interaction with P3.)

P2) Once the next candidate point is chosen the parameters of the fit must be updated. (Not too hard, usually efficient recursive updating is possible for practical models.)

P3) By using the likelyhood value for the fit (or its derivative) it must be decided if encompassing the new point is a good alternative. (This so called stopping rule is a major problem because it involves the interaction of models at a joint.

It may be that the traditional hypothesis testing via curve fitting, which seems at first to be the natural thing to do, is really too involved and hence frail in lieu of other alternatives. Making cruder detections via Hough techniques is discussed next in Section 5.2 and Section 5.3 presents a highly practical alternative in cases where the curve shape is highly constrained a priori.

5.2 Hough detection/verification of other curves.

Stereotyped curve structures such as segments of ellipses, parabolas, hyperbolas, and circles can be detected using the Hough transform. Because it is essentially template matching, Hough detection will be robust when a large number of points lie on the curve segment. Because an entire set of parameterized templates are used at once variation in the curve from an a priori shape is permitted and a best set of parameters can be chosen.

The parametric forms

$$x^2 + axy + by^2 + cx + dy + e = 0$$

$$\text{or} \quad a x^2 + bxy + y^2 + cx + dy + e = 0$$

can be used to specify any of the forms - - circles, ellipse, parabola, or hyperbola [Rees 1963]. Estimating five parameters, however, will be expensive and frail even with the Hough technique. Small segments of all the above mentioned curves might be approximated by a circular arc reducing the number of parameters to only 3. In their work on the recognition of manufactured parts Tsuji and Matsumoto [1978] were successful in detecting elliptical curve segments by using a hierarchical approach combining Hough detection with curve fitting. In a first stage 2 center parameters are detected using the Hough transform. Given a chosen center a second Hough detection stage determines a third parameter. All points consistent with the 3 parameterschosen in the first two stages are then fit with a full 5 parameter elliptical model to determine the final detection decision. Finally the actual extent of the curve must be extracted from the set of all points on the full elliptical template.

5.2.1 Hough circle detection

A circle $(x-x_c)^2 + (y-y_c)^2 = r^2$ is specified by the three parameters (x_c,y_c,r). Kimme et al [1975] have presented a practical method for Hough circle detection which includes use of gradient information for sharpening the transform and ultimate extraction of points on the detected curve. A Hough circle detection algorithm is presented below. The effective parameter space is highly constrained by the verification hypothesis. High gradient points are then selected only from a restricted region. Each possible circle is represented by one accumulator or bin which essentially defines a template or mask laid on the image. Figure 42 shows 3 such templates. For each selected high gradient point it is recorded which hypothetical circles are supported by the point evidence. If the point evidence is consistent with the hypothetical circle the appropriate bin is incremented. For example, the point evidence (x,y,d) in Figure 42 supports the existence of circle (x_3,y_3,r_3) but not circles (x_1,y_1,r_1) or (x_1,y_1,r_2).

```
Begin Hough circle detection
    . Initialization
        Select a set S of high gradient points from a certain area
        of the image. $S = \{(x_i,y_i,d_i)\}, i=1,N$ where $d_i$ is the direction
        of the gradient at $(x_i,y_i)$
        Establish an empty set of accumulator arrays for the discretized (x,y,r)-parameter space where r is the radius of a circle with center at (x,y). (The variation of parameters x,y and r are highly constrained by the hypothesis being verified.)

    . Distribution of local evidence

        For each point $(x_i,y_i,d_i)$ in S do block A

            For each possible center $(x_c,y_c)$ compatible with $(x_i,y_i)$ and direction $d_i$ do block B

                Compute $r = ((x_i-x_c)^2 + (y_i-y_c)^2)^{1/2}$.
                Distribute mass to $(x_c,y_c,r)$ in set of accumulators.

            end block B

        end block A

    . Detection of global structure
        Smooth the accumulator array. Detect above threshold accumulator array peaks.

end Hough circle detection
```

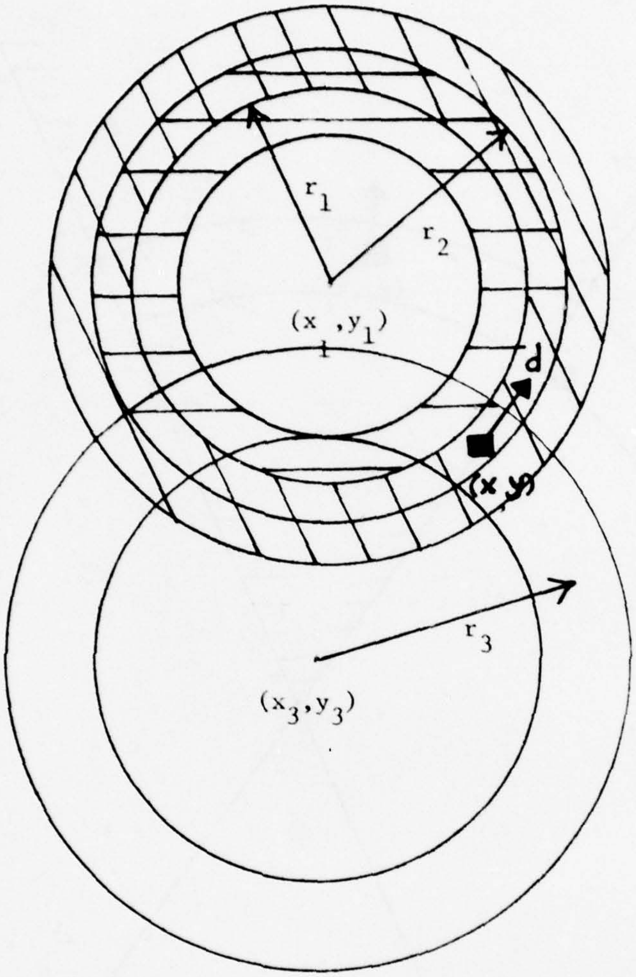


Figure 42. Three templates used in circular Hough detection. (The parameterizations are (x_1,y_1,r_1),(x_1,y_1,r_2) and (x_3,y_3,r_3).)

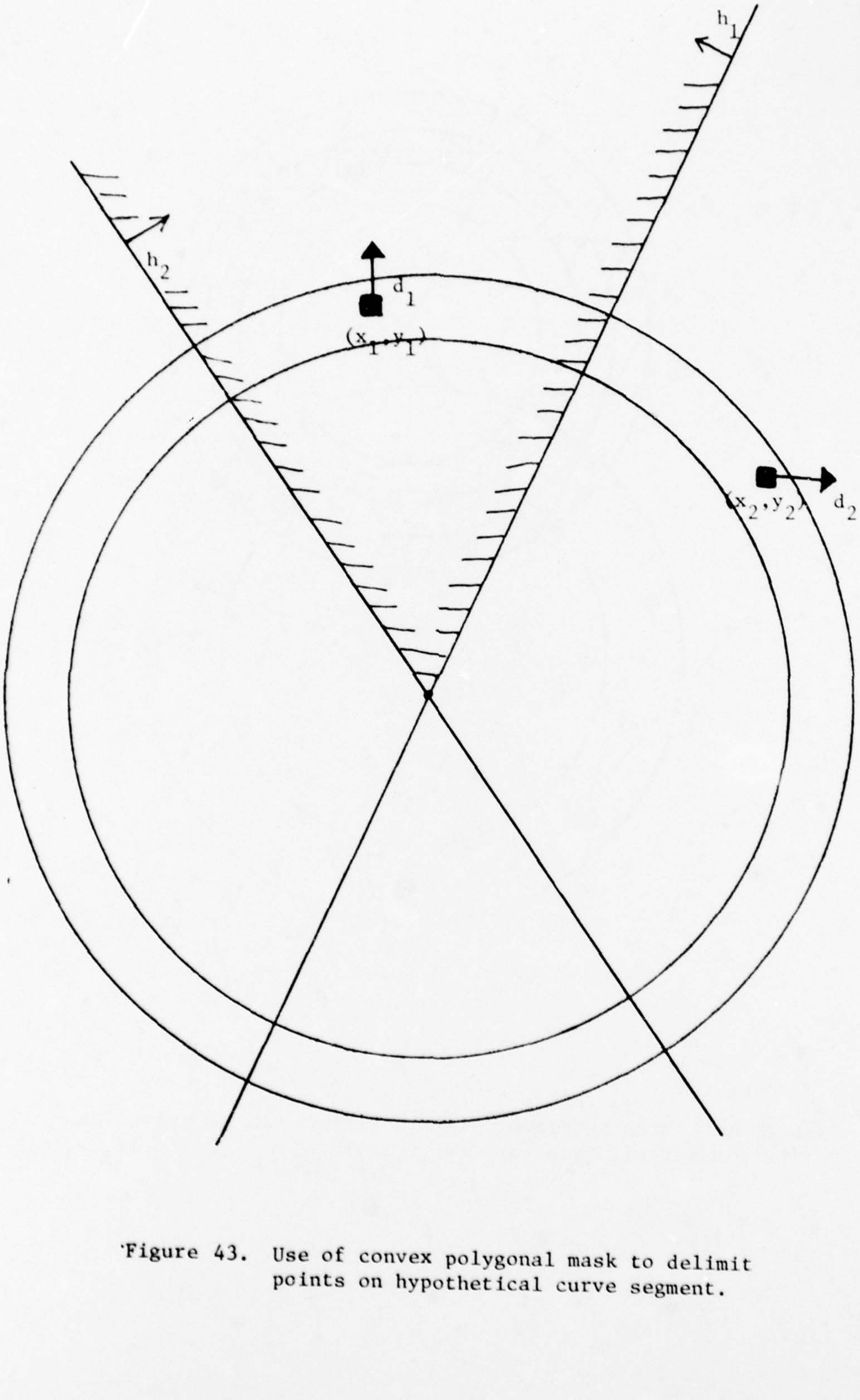


Figure 43. Use of convex polygonal mask to delimit points on hypothetical curve segment.

5.2.2 Selection of point evidence

In order to detect segments (arcs) of a mathematical curve (circle, line, etc.) it will be necessary to restrict the template used to some subset of its original coverage. This is perhaps best done, not by changing the parameterization, but by masking the region from which points are selected. For an example consider Figure 43. Verification of the existence of a circular arc in the shaded region is desired. The shaded region is determined by the circular (actually ring-shaped) template intersected with two halfplanes h_1 and h_2. Points selected as evidence can easily be confined to any convex polygon of n sides by application of n linear inequalities (halfplane tests). Any accumulated evidence for global structure is then guaranteed to be dense in a restricted region rather than being disperse in a complete template.

5.3 Verification of particular boundary curves

When searching the data for a particularly shaped boundary or linear feature neither curve-fitting nor Hough detection are appropriate. Not only might there be too many parameters for robust detection but also there is no need to bear the greater expense relative to the practical alternative given below. Here we are talking about verification of features whose shape is precisely known such as the path of a stream or the wing of a certain airplane.

5.3.1 A practical verification technique for particular curves

As a first example consider the verification of the existence of Sherman Creek in imagery of Harrisburg, Penna. (refer to Figure 17). It is easy to store the path of the stream as an ordered set of points in some coordinate system as is done in cartographic data bases. Given a registration of imagery containing the creek to the cartographic coordinate system, it is easy to transform the points of the mapped feature to the points (pixels) of the image where the feature should be found. Due to noise, distortion, approximation in the registration transformation, and actual change in the stream, it is unlikely that the feature points will be found exactly where they are predicted to be. Thus some tolerance must be used in the verification and the verification result should be an evaluation of how far the detected feature points deviated from their predicted position. A root-mean-square value is one measure of the match. Let $P = \{p_1, \ldots, p_m, \ldots, p_n\}$ be the feature point set in map coordinates and let T_α be the transformation registering the imagery to the map coordinate system. Let q_m be the best detection of point p_m in the neighborhood of $T_\alpha^{-1}(p_m)$ in the image. Then one measure of the verification of point set P is

$$D(P) = ((\sum_{m=1}^{N} d^2(T_\alpha^{-1}(p_m), q_m))/N)^{1/2}$$

where $d^2(\cdot,\cdot)$ is the squared distance between the two image points. Not only did Barrow et al [1977] use such a verification measure, but they also did hill-climbing on the parameter set α in order to get a better registration T_α^{-1} .

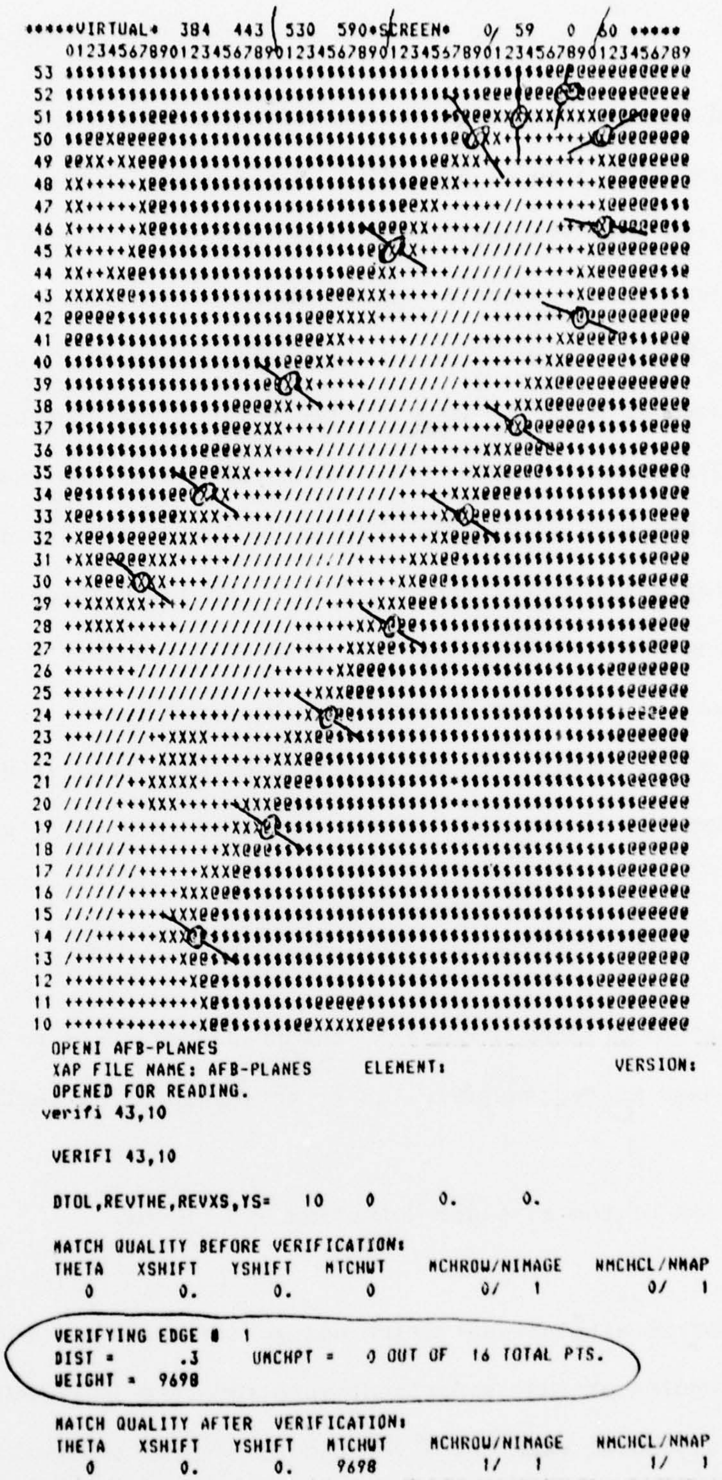
As a second example, consider the verification of the nose of an airplane after registration with a model according to the procedures of Section 4. Points to be tested on the nose are appended to the model as a list P = $\{ p_1,...,p_n\}$ defining the boundary curve. Once T_α is determined as in Section 4, T_α^{-1} can be used to define where the curve should be in the image. The image is searched along a normal to the curve at point $T_\alpha^{-1}(p_m)$ for the point q_m of highest gradient magnitude and compatible gradient direction. The match value e(p) can then be evaluated. Figure 44 shows the nose of the plane in image AFB1. 16 mapped points $T_\alpha^{-1}(p_m)$ (circled) defining a nose are transformed into the image space and the gradients are examined along profiles normal to the predicted boundary curve. Points q_m of the detected boundary curve are selected on these profiles. In the case shown in Figure 44 D(P) = 0.3 pixels. The RMS distance between predicted and detected positions can be compared to a tolerance and converted to a match value in the range [0,1].

$$e(P) = \max \{ 0,1- \frac{D(P)}{DTOL} \}$$

The method of searching along normals to the predicted curve to detect feature points has been used by Perkins[1977] in a parts inspection application.

5.3.2 Continuation of the airplane detection experiment

An experiment in airplane detection using straight edge evidence was described in Section 4. Results of that experiment were reported in Column 3 of Table 3 and showed that not all model edges were detected. Several ambiguities existed due to considerable rotational symmetry at 90°,180°, and 270° off of the true airplane



```
*****VIRTUAL*  384  443  530  590*SCREEN*   0  59   0  60 *****
   012345678901234567890123456789012345678901234567890123456789
53 $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@@@@@@@@@
52 $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@@@@@@@@@
51 $$$$$$$$@@@$$$$$$$$$$$$$$$$$$$$$$$$$$+@@XXXXXXXXXXXX@@@@@@@@
50 $$@@X@@@@@@$$$$$$$$$$$$$$$$$$$$$$$$$$@XXX++++++++++X@@@@@@@@
49 @@XX+XX@@@$$$$$$$$$$$$$$$$$$$$$$$$$@@XXX+++++++++++XX@@@@@@@
48 XX+++++X@@$$$$$$$$$$$$$$$$$$$$$$$@@@XX+++++++++++++X@@@@@@@@
47 XX+++++X@@$$$$$$$$$$$$$$$$$$$$$$@@XX++++++//+++++++X@@@@@$$$
46 X++++++X@@$$$$$$$$$$$$$$$$$$$@@@XX+++++///////+++++X@@@@@@$$
45 X+++++X@@$$$$$$$$$$$$$$$$$$$$@@XX+++++///////+++++X@@@@@@@@@
44 XX++XX@@$$$$$$$$$$$$$$$$$$$@@@XX+++++///////+++++XX@@@@@@@$@
43 XXXXX@@$$$$$$$$$$$$$$$$$$@@@XXX+++++///////++++++X@@@@@@$$$$
42 @@@@@$$$$$$$$$$$$$$$$$$@@@XXXX+++++/////+++++++++@@@@@@@@@@@
41 @@@$$$$$$$$$$$$$$$$$$$@@@XX++++++//////++++++++XX@@@@@@$$@@@
40 $$$$$$$$$$$$$$$$$$$$$@@@XX+++++////////+++++++XX@@@@@@$@@@@@
39 $$$$$$$$$$$$$$$$$$$@XXX+++++//////////++++++XXX@@@@@@@@@@@@@
38 $$$$$$$$$$$$$$$$@@@@XX+++++/////////++++++XXX@@@@@@@$$$@@@@@
37 $$$$$$$$$$$$$$$@@@XXX++++//////////+++++++X@@@@@@$$$$$$$@@@@
36 $$$$$$$$$$$$$@@@@XXX++++//////////++++++XXX@@@@@$$$$$$$@$@@@
35 @$$$$$$$$$$@@@@XXX++++///////////++++++XXX@@@@@$$$$$$$$@@@@@
34 @@$$$$$$$$@@XXX+++++////////////+++++XXX@@@@@$$$$$$$$$$$@@@@
33 X@@$$$$$$@@XXXX+++++///////////++++++XX@@@@$$$$$$$$$$$$@@@@@
32 +X@@$@@@@XXX++++/////////////++++++XX@@@$$$$$$$$$$$$$$$@@@@@
31 +XX@@@@@XXX+++++////////////+++++XXX@@$$$$$$$$$$$$$$$$$$@@@@
30 ++X@@@XXX++++//////////////+++++XX@@@$$$$$$$$$$$$$$$$$$@@@@@
29 ++XXXXXX++++/////////////+++++XXX@@@@$$$$$$$$$$$$$$$$$$@@@@@
28 ++XXXX+++++////////////++++++XX@@$$$$$$$$$$$$$$$$$$$$$$@@@@@
27 +++++++++/////////////+++++XXX@@$$$$$$$$$$$$$$$$$$$$$$$@@@@@
26 +++++++/////////////++++++XX@@@$$$$$$$$$$$$$$$$$$$$$@@@@@@@@
25 ++++++/////////////+++++XXX@@@$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@
24 ++++//////++++++/++++++XX@@$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@
23 +++//////++XXXX+++++++XXX@@$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@@
22 ///////++XXXX+++++++XXX@@$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@@
21 //////++XXXXX+++++XXX@@@$$$$$$$$$$$$$+$$$$$$$$$$$$$$$$@@@@@@
20 /////+++XXX+++++XXX@@$$$$$$$$$$$$$$$$+$$$$$$$$$$$$$$$$$@@@@@
19 /////+++++++++++XX@$$$$$$$$$$$$$$$$$$$+$$$$$$$$$$$$$$$$@@@@@
18 //////+++++++++XX@@@@$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@
17 ///////++++++XXX@@$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@@
16 //////+++++XXX@@@$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@@
15 /////++++XXX@@$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@
14 ///++++++XX@$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@
13 /++++++++++X@@$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@
12 ++++++++++++X@@$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$@@@@@@@@
11 +++++++++++++X@$$$$$$$$$@@@@@$$$$$$$$$$$$$$$$$$$$$$$@@@@@@@@
10 +++++++++++++X@@$$$$$$$$@@XXXX@@$$$$$$$$$$$$$$$$$$$$$$@@@@@@
  OPENI AFB-PLANES
  XAP FILE NAME: AFB-PLANES      ELEMENT:                   VERSION:
  OPENED FOR READING.
 verifi 43,10

  VERIFI 43,10

  DTOL,REVTHE,REVXS,YS=   10    0      0.      0.

  MATCH QUALITY BEFORE VERIFICATION:
  THETA    XSHIFT    YSHIFT   MTCHUT    MCHROU/NIMAGE   NMCHCL/NMAP
    0         0.        0.       0           0/   1          0/   1

  VERIFYING EDGE #   1
  DIST =       .3     UMCHPT =   0 OUT OF  16 TOTAL PTS.
  UEIGHT =  9698

  MATCH QUALITY AFTER  VERIFICATION:
  THETA    XSHIFT    YSHIFT   MTCHUT    MCHROU/NIMAGE   NMCHCL/NMAP
    0         0.        0.    9698           1/   1          1/   1
```

Figure 44. Verification of the existence of airplane nose in AFB1 image.

position. In particular for plane Number 1 there were strong responses at both $\theta=145°$ and $\theta=329°$. In a second phase of the plane detection experiment model edges that were not detected in the original bottom-up feature detection were verified top-down under the possible registration transformations. For the case in point in Table 4.3 the transformation ($\theta=329°$,xs=-108,ys=46) explained only 8 of the 10 model edges with image edge evidence while the transformation ($\theta=145°$, xs=133,ys=-36) accounted for only 6 of 10 model edges. It was expected that verification would strongly enhance the object detection results and eliminate ambiguities. In particular verification of the tail of the plane, whose edges were often missed by primary detection, would eliminate ambiguities caused by rotational symmetry.

The experimental procedure was as follows. The results of registration (in Column 3 of Table 3) were written out to a file for input to the verification. Along with each set of possible registration parameters α was recorded exactly which model edges were not explained by the primary detections and the transformation T_α. This output was then used in the verification phase to perform focused searches in the imagery for previously undetected edges. The previously derived match values for each T_α were then adjusted according to the verification evidence.

5.3.3 Results of verification of airplane detections

Table 6 contains a summary of the results of verification of the initial airplane detections listed in Table 3. In every case the verification process was able to significantly increase the confidence in the detection decision. In subwindows 1 and 7, where no planes were present, no new positive evidence was gathered. In fact failure to find any additional edge evidence resulted in large drops in match weight, to 3.4 and 11.1 respectively on a scale of 100. For the five subwindows which contained planes the match confidence went up for the correct detection and went down for the incorrect alternative detection. In each case all 10 model edges were found for the correct detection while exactly 7 edges were found for the alternatives. The 3 edges of the tail were not found in all the alternative detections. It should be noted how important the tail is in countering the rotational symmetry of the plane and in differentiating an airplane from an intersection of roads. Additional verification of points on the nose, engines, wing and tail tips would give further confidence to the detection decision. Match weights for correct matches of primary detections were in the range of 40 to 60 out of a possible 100, primarily because the Hough detected edge elements either undershot or overshot the true endpoints of the edge. Match weights for verified edges were roughly in the same range due to slight misalignment of predicted positions and detected positions on the edge. Hough detections could be cleaned up to remove the first "problem" and the alignment T_{α}^{-1} could be adjusted to remove the second. Although the code for performing such adjustments was actually at hand, it was decided not to complicate the experiments by using it. The results obtained so far are conclusive enough to demonstrate the viability of both the registration and verification.

TABLE 6

RESULTS OF VERIFICATION APPLIED TO REGISTRATION HYPOTHESES OF TABLE 3.

Center of Sub-Window	True Transformation obtained interactively (± standard deviation) θ XSHIFT YSHIFT	Results of registration with primary detection of straight edges						Results adjusted by secondary verification			
		θ	xs	ys	- evaluation -				- evaluation -		
1 (256,256)	No planes	204	-110	56	17.10	3/53	2/10		3.4	3/53	2/10
2 (256,512)	Plane 1 328±1,-107±3,47±2	329	-108	46	42.20	9/40	8/10	+	48.9	11/40	10/10
		145	133	-36	45.10	6/40	6/10	&	30.7	7/40	7/10
3 (256,768)	Plane 2	327	-134	-16	35.30	10/40	9/10	+	38.7	11/40	10/10
	329±1,-132±2,-19±2	57	34	-145	25.10	6/40	5/10	&	16.5	8/40	7/10
4 (256,1024)	Two planes (2&3)	330	-159	-89	52.30	9/48	8/10	+	54.2	11/48	10/10
	2) 329±1,1±2,200±4	240	-87	178	33.20	7/48	6/10	&	27.1	8/48	7/10
	3) 328±1,-162±2,-84±2										
5 (256,1280)	Two planes (3&4)	330	-181	-151	40.10	7/48	8/10	+	43.6	9/48	10/10
	3) 328±1,-26±3,133±2	56	-110	-24	20.80	7/48	6/10	&	19.4	8/48	7/10
	4) 330±1,-184±2,-152±2										
6 (256,1536)	Plane 4	332	-59	66	63.30	18/70	9/10	+	64.6	19/70	10/10
	330±1,-55±2,69±2	153	77	-51	40.70	10/70	7/10	&	28.2	10/70	7/10
7 (256,1792)	No planes	268	-49	12	27.80	4/44	4/10		11.1	4/44	4/10

& tail of plane not found

+ all model edges of airplane found

6. Conclusions

This report studied the use of models in image analysis. Models are essentially structured a priori information which can be used to interpret data in a manner consistent with global real world knowledge. Potential models are evoked by features extracted by primitive bottom-up processing. Features studied in this research were all derivable from boundary curve segments and included straight edge segments, circular or parabolic edge segments, and points of high curvature on boundary segments. Two types of models were considered. One type of model was the formal Problem Reduction Representation (PRR) or Context Free Grammar (CFG). The second type of model was the iconic model, which corresponded to a cartographic data base, in which the form of particularly shaped geographic features was encoded. Regardless of the type of model being used, primitive features are used to allign a hypothetical model with raw image data. All further image analysis is then made by verification (or denial) of hypotheses generated (top-down) from the model. Verification of shape features in the imagery was viewed as either curve-fitting under constraints or template (icon) matching.

The study attempted to draw conclusions about an entire system by studying its several possible parts. Many experiments were performed and a great deal of literature was reviewed. As a result of this study the following conclusions have been reached.

- Totally bottom-up synthesis of an image interpretation is at best inefficient and at worst logically impossible. To form an interpretation that is to be useful is to integrate an instance of data with a large amount of stored knowledge. During this process it is possible that some of the physical evidence will be ignored or even contradicted.

- Representing real world knowledge, particularly for use by a computer, is a difficult task with much current research activity. Use of general shape models such as PRR or CFG has shown some promise but often turns out difficult in practice. In the current study the use of particular icons, i.e. maps, appeared to be much more practical.

- Current automatic low level feature detection techniques can support complex analysis when features are registered to an iconic (geographic/cartographic) data base. The registration process must consider the feature set as errorful and partial, yet be able to arrive at correct decisions. A clustering method for arriving at correct global interpretations from errorful and partial local evidence was developed during the period of study.

- The clustering technique for making global decisions from imperfect evidence has advantages over sequential syntactic procedures and iterative relaxation procedures because multiple competing global hypotheses can be simultaneously weighed. Incorrect or ambiguous local structure can not propagate because all evidence is considered in parallel (i.e. order independent).

- While the several heuristics used in this study for evaluation of competing interpretations seemed to work in practice, the determination of confidence in a hypothesis from combined evidence remains a sticky problem in theory and in practice. In this study only shape features were used, but in a practical system shape information will be used with say color and non-imagery derived intelligence data. Much more work and thought is needed to study the problem of making decisions on combined evidence.

- The most promising future direction for reconnaissance image analysis appears to be along the line of map-guided image analysis. A huge resource of iconic real world knowledge already exists in cartographic data bases. This study showed how that resource might be used in automatic or semiautomatic image analysis. Further work in this area is likely to advance the encoding of geographic data bases to enhance the automatic image analysis function.

References

Ashbaugh, J.B.,1973, The Detection and Identification of Man-Made Objects from Aerial Reconnaissance Photographs, NTIS (AD-760 755).

Barrow, H. G., R. C. Bolles, T. D. Garvey, J. H. Kremmers, K. Lantz, J. M. Tenenbaum and H. C. Wolf, Oct. 1977, Interactive Aids for Cartography and Photointerpretation Progress Report, Image understanding workshop proceedings, Palo Alto, Calif., Applications, Inc. Report SAI-78-656-WA, Lee S. Baumann, editor.

Barrow, H. G., J. M. Tenenbaum, R. C. Bolles and H. C. Wolf, Aug. 1977, Parametric Correspondence and Chamber Matching, Two new techniques for image matching, Proceedings of the 5th International Joint Conference on Artificial Intelligence, M.I.T.

Bullock, B., Oct. 1974, The Performance of Edge Operators on Images with Texture, Hughes Research Laboratory Tech. Report.

Chang, C. L. and J. R. Slagle, 1971, An admissible and optimal algorithm for searching AND/OR graphs, Artificial Intelligence, Vol. 2, pp 117-128.

Chartres, B.A., J.J. Florentin, July 1968, A universal syntax-directed top-down analyzer, C.A.C.M., Vol. 15, No. 3, pp 447-464.

Cooper, D. B., 1976, On the Recognition of Highly Variable Line Drawings through Use of Maximum Likelihood Functions in Pattern Recognition and Artificial Intelligence, C.H. Chen (ed.), Academic Press, Inc. pp 145-163.

Davis, L., Sept 1975, A Survey of Edge Detection Techniques, Computer Graphics and Image Processing, Vol. 4, No. 3, pp 248-270.

Duda, R. O. and P. E. Hart, 1973, Pattern Classification and Scene Analysis, Wiley and Sons, N.Y.

Duda, R. and P. Hart, Jan. 1972, Use of the Hough transformation to detect lines and curves in pictures, C.A.C.M., Vol. 15, No. 1, pp 11-15.

Dudani, S.A., A. L. Luk, P.P. Stapudd, C. S. Clark and B. L. Bullock, July 1977, Model-Based Scene Matching, Hughes Research Report #509

Fu, K.S., 1974, Syntactic Methods in Pattern Recognition, Academic Press.

Griffith, A.K., 1973, Edge detection in simple scenes using a priori information, IEEE TC-22, No. 4, pp 371-380.

Guzman, A., 1971, Analysis of curved line drawings using context and global information, Machine Intelligence 6, pp 335-376.

Hall, P.A.V., 1973, Equivalence between AND/OR graphs and context-free grammars, C.A.C.M., Vol. 16, No. 7, pp 444-445.

Horn, B.K.P. and B. L. Bachman, Aug. 1977, Using Synthetic Images to Register Real Images with Surface Models, M.I.T. A.I. Memo 437, Mass. Institute of Technology, Cambridge, Mass.

Kimme, C., D. Ballard and J. Sklansky, Feb. 1975, Finding circles by an array of accumulators, Comm. A.C.M., Vol. 18, No. 2, pp 120-122.

Ledley, R., 1966, Pattern Recognition Studies in the Biometrical Sciences, AFIPS Conf. Proc. SJCC, p 411.

Marr, D., 1975, Early processing of visual information, A.I. Memo 340, Artificial Intelligence Lab., M.I.T.

Martelli, A., 1976, An application of heuristic search methods to edge and contour detection, C.A.C.M., Vol. 19, No. 2, pp 73-83.

Miller, P.L., 1973, A locally organized parser for spoken output, Lincoln Lab., M.I.T., Cambridge, Mass., Tech. Report 503.

Nilsson, N.J., 1971, Problem Solving Methods in Artificial Intelligence, McGraw-Hill Book Co.

Pavlidis, T., 1977, Structural Pattern Recognition, Springer-Verlag, New York.

Perkins, W. A., Aug. 1977, Model-based vision system for scenes containing multiple parts, Proceedings of the 5th International Conf. on Artificial Intelligence, M.I.T., Cambridge, Mass., pp 678-684.

Ralston, A., 1965, A First Course in Numerical Analysis, McGraw-Hill Book Co., N.Y.

Rees, P., 1963, Analytical Geometry, Prentice-Hall, Inc. Englewood Cliffs, N.J.

Rosenfeld and A. Kak, 1976, Digital Picture Processing, Academic Press, N.Y.

Rosenfeld and Thurston, May 1971, Edge and Curve Detection for Visual Scene Analysis, IEEE Transactions on Computers, Vol., C-20, No. 5, pp 562-569.

Shaw, A. G., July 1970, Parsing of Graph-representable pictures, JACM, Vol. 17, pp 453-481.

Stockman, George C. and Laveen N. Kanal, June 1976, Interactive Screening of Reconnaissance Imagery, AMRL-TR-76-15 (AD A-028969), Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Ohio.

Stockman, George C. July 1978, Toward Automatic Extraction of Cartographic Features, ETL-0153, U. S. Army Engineer Topographic Laboratories, Ft. Belvoir, Va.

Stockman, G. C., May 1977, A problem-reduction approach to the linguistic analysis of waveforms, Ph.D. dissertation, Univ. of Md., Computer Science Dept., College Park, Md.

Stockman, G. C. and A. Agrawala, Nov. 1977, Equivalence of Hough Curve Detection to Template Matching, C.A.C.M., Vol. 20, No. 11, pp 820-822.

Stockman, G. C., Nov. 1978, Defining and Extracting Waveform Primitives for Linguistic Analysis, Proceedings 4th Internation Joint Conf. on Pattern recognition, Kyoto, Japan.

Tenenbaum, J. M., 1976, IGS: A paradigm for integrating image segmentation and interpretation, Pattern Recognition and Artificial Intelligence, C.H. Chen (ed.) Academic Press, pp 472-507.

Tsuji, S. and F. Matsumoto, Aug. 1978, Detection of Ellipses by a Modified Hough Transformation, IEEE-TC, Vol. C-27, No. 8, pp 777-780.

VanderBrug, G. and J. Minker, 1975, State-space, problem-reduction, and theorem proving-some relationships, C.A.C.M., Vol. 18, No. 2, pp 107-115.

VanWie, Peter and Maurice Stein, July 1977, A Landsat Digital Image Rectificaion System, IEEE Trans Geoscience Electronics, Vol. GE-15, No. 3.

Zahn, C.T., 1974, An Algorithm for Noisey Template Matching, Proc. IFIP Congress, pp 698-701

Zahn, C. T. and R. Z. Roskies, Mar. 1972, Fourier Descriptors for Plane Closed Curves, IEEE-TC-21, No. 3, pp 268-281.

Zucker, S. W., 1976, Relaxation Labelling, Local Ambiguity and Low-level Vision, Pattern Recognition and Artificial Intelligence, C.H. Chen (ed.), Academic Press.

☆U.S.Government Printing Office: 1979 — 657-002/442